AMERICAN JOURNAL OF MANAGEMENT AND IOT MEDICAL COMPUTING
Peer Reviewed, Referred & Indexed Journal
E-ISSN: 3069-0110
Vol.5, No.2(2026)
www.ajmimc.com

Detecting Anomalous Patterns in IoT Healthcare Systems through Context-Aware Intelligence

V. Bharathi[1], Sk. Kashif[2], K. K. C. Preetam Vignesh[2], E. Sujin[2], T. Sai Tanay Babu[2]

[1]Associate Professor, [2]UG Student, [1,2]Department of Computer Science and Engineering (AI & ML)

[1,2]Geethanjali Institute of Science and Technology, Nellore-Bombay Highway, S.P.S.R, Andhra Pradesh 524137, India

Abstract

The rapid growth of smart healthcare systems and interconnected environments has increased the demand for secure, scalable, and intelligent data analysis mechanisms. With the emergence of distributed architectures and cloud-based services, achieving accurate prediction and efficient anomaly detection has become a major challenge. This research presents a cloud-driven intelligent system that integrates machine learning (ML) techniques with a client–server architecture for secure and real-time prediction. The primary problem addressed in this work is the difficulty in obtaining high prediction accuracy from large and imbalanced datasets in distributed environments. Many traditional systems rely on standalone models and manual analysis, resulting in limited accuracy, poor scalability, and ineffective handling of class imbalance. These limitations reduce their suitability for real-time and large-scale applications. There is a strong need for an automated, robust, and scalable system capable of performing efficient data preprocessing, handling imbalance, supporting multiple ML models, and enabling secure remote access. The system should also ensure seamless communication between distributed components. To address these challenges, the proposed system adopts a client–server architecture consisting of Laptop 1 (LP1) as the server and Laptop 2 (LP2) as the client. LP1 integrates a desktop graphical user interface with a cloud-based prediction engine using the Flask framework. The system performs preprocessing using label encoding (LE) and applies the Synthetic Minority Oversampling Technique (SMOTE) for class balancing. Multiple models including Ridge Classifier (RC), Quadratic Discriminant Analysis (QDA), and Perceptron (PC) are implemented. Furthermore, a Locally Deep Ensemble Classifier (LDEC) combines Histogram Gradient Boosting Classifier (HGB) and Light Gradient Boosting Machine Classifier (LGBM) using soft voting (SV). Secure authentication uses Redis. The system improves accuracy, scalability, reliability, and security for modern applications.

Keywords: Cloud Computing, Smart Healthcare Systems, Machine Learning, Client–Server Architecture, Real-Time Prediction, Anomaly Detection, Imbalanced Data, Data Preprocessing.

1. INTRODUCTION

The rapid growth of the Internet has driven the evolution of the Internet of Things (IoT), enabling the development of intelligent and interconnected environments such as smart cities, smart homes, and industrial automation systems [1]. Within this expanding ecosystem, the integration of advanced medical sensors, wearable devices, and remote monitoring systems has given rise to the Internet of Medical Things (IoMT), which plays a vital role in modern healthcare by supporting real-time patient monitoring, early diagnosis, and efficient clinical decision-making. However, the increasing connectivity and continuous data exchange in IoT and IoMT environments introduce significant security challenges, particularly due to the transmission of sensitive healthcare information across distributed networks. As a result, ensuring data confidentiality, integrity, and availability has become a critical requirement in these systems. Intrusion Detection Systems (IDSs) serve as an essential

security layer in such environments, capable of identifying malicious activities and unauthorized access attempts that may not be detectable through traditional security mechanisms [2]. IDS solutions can be implemented in both software and hardware forms and are designed to monitor network traffic, analyze behavioral patterns, and detect anomalies in real time. In the healthcare domain, where system reliability and data accuracy are crucial, the importance of robust IDS frameworks becomes even more significant. The rapid adoption of IoT-based healthcare technologies has improved operational efficiency and patient care quality, but it has also expanded the attack surface, making systems more vulnerable to cyber threats such as data breaches, denial-of-service attacks, and unauthorized access [3].

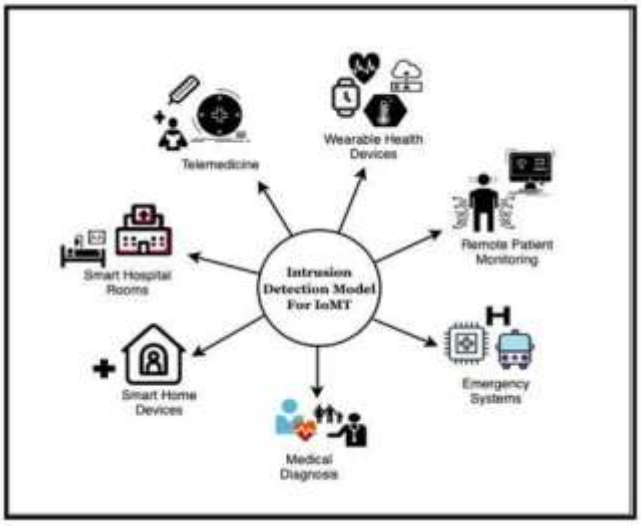


Fig. 1: Intrusion Detection System for Securing Internet of Medical Things

To overcome these limitations, ensemble learning techniques have emerged as a powerful approach for enhancing IDS performance. Ensemble methods such as bagging, boosting, and stacking combine multiple base models to achieve higher prediction accuracy, improved generalization, and better resilience against noise and imbalanced data [4]. As illustrated in Fig. 1, ensemble-based IDS frameworks leverage diverse learning algorithms and integrate their outputs using meta-learning strategies, enabling more accurate and stable intrusion detection. This approach reduces false alarm rates and enhances the system's ability to detect both known and unknown attack patterns. Furthermore, the integration of ensemble learning into IDS systems allows for adaptive and dynamic decision-making, which is essential in highly complex and evolving IoT and IoMT environments. By incorporating techniques such as feature optimization, model diversity, and decision fusion, ensemble-based IDS can effectively handle varying network conditions and large-scale data streams [5].

2. LITERATURE SURVEY

Mousa Alalhareth et, al. [6] proposed a fuzzy-based self-tuning Long Short-Term Memory (LSTM) intrusion detection system (IDS) for the IoMT. Their approach dynamically adjusts the number of epochs and utilizes early stopping to prevent overfitting and underfitting. They conducted extensive experiments to evaluate the performance of their proposed model, comparing it with existing IDS models for the IoMT. Arash Salehpour, et al. [7] Proposed the system that decreases the dimensionality by operating the MI filtering on the dataset and keeps only the most informative features. It further refines this using ensemble-based ranking methods, such as Random Forest, AdaBoost, XGBoost, and LightGBM, to ensure the optimum feature selection for the task. Random

Forest was adopted for the final classification because it is generally robust, efficient at handling high-dimensional data, and usually performs well. The proposed system has been tested with intensive usage using three well-acknowledged benchmark datasets, namely WUSTL-EHMS-2020, NSL-KDD, and CIC-IoMT2024. It showed considerable accuracy, precision, recall, and F1-score gains, particularly for DDoS and DoS attack types. Nikhil Sharma et, al. [8] The proposed DA-DRL-AES-SHA-512 methodology significantly outperformed conventional encryption techniques, achieving an encryption time of 0.0975 s, decryption time of 0.0846 s, and a throughput of 75.63 transactions per second (Tx/s) with a network overhead of just 0.1289%. The Energy consumption and computational overhead are reduced to 0.3664 J and 0.48%, respectively. The Secure and Dependable Bi-LSTM GRU Intrusion Detection Framework (S-BiLSTMGRU-IDF) achieved 99.94% accuracy in binary classification and 99.89% in multiclass classification, improving detection efficiency by 0.6–3.5% over state-of-the-art models. Georgios Zachos et, al. [9] Introduced an AIDS specifically designed for resource-constrained devices within IoMT networks. The proposed lightweight AIDS leverages novelty detection and outlier detection algorithms instead of conventional classification algorithms to achieve (a) enhanced detection performance against both known and unknown attack patterns and (b) minimal computational costs.

Abdelatif Hafid et, al [10] proposed a high-performance cybersecurity framework leveraging a carefully fine-tuned XGBoost classifier to detect malicious attacks with superior predictive accuracy while maintaining interpretability. Their comprehensive evaluation compared the proposed model with a well-regularized Logistic Regression baseline using key performance metrics. Additionally, they analyze the security-cost trade-off in designing ML systems for threat detection and employ SHAP (SHapley Additive exPlanations) to identify key features driving predictions. They further introduced a late fusion approach based on max voting that effectively combines the strengths of both models. Faeiz Alserhani et, al [11] The system is constructed on a feedback-looped architecture integrating hybrid feature modeling, physical behavioral analysis, and Extreme Learning Machine (ELM)-based classification to provide adaptive access control, continuous monitoring, and reliable intrusion detection. ML-CCPS is capable of outperforming benchmark classifiers with an acceptable computational cost, as evidenced by its macro F1-score of 97.8% and an AUC of 99.1% when evaluated with the ToN-IoT dataset. Mohammad Zubair Khan et, al. [12] introduced a novel hybrid anomaly detection model combining a Graph Convolutional Network (GCN) with a transformer architecture. The GCN captures the structural relationships within the IoMT data, while the transformer models the sequential dependencies in the anomalies

Arezou Naghib et, al [13] paper identified 28 critical studies published between 2018 and April 2024. The intrusion detection mechanisms in the IoMT are divided into five categories: IDS based on artificial intelligence models, datasets used in IoMT for IDS, fundamental security requirements, intrusion detection processes, and evaluation metrics. This paper dissects the various mechanisms within each category in a meticulous and comprehensive analysis. Yahya Rbah et, al. [14] presented a low-cost, high-accuracy ML-based attack detection framework for securing IoMT devices. Eight ML models, including Decision Tree, Support Vector Machine, Naive Bayes, Gradient Boosting, K-Nearest Neighbor, Random Forest, and XGBoost, were evaluated on the IoT-Healthcare security dataset. XGBoost emerged as the top performer, achieving 99.98% accuracy in just 233 ms. Jordi Doménech et, al. [15] addressed these limitations by comparing the performance of ML models trained on a general IoT dataset (CICIoT2023) and an IoMT-specific dataset (CICIoMT2024) to demonstrate the importance of domain-specific data. Their findings reveal substantial drops of up to 66.87% in the F1-score when models trained on one dataset are tested on the other.

3. PROPOSED SYSTEM

The proposed User Cloud-Driven Intrusion Detection System (IDS) is designed to enhance the security of Internet of Medical Things (IoMT) environments by integrating advanced machine learning techniques with a cloud-based architecture. The system follows a dual-module design consisting of Laptop 1 (LP1) as the server and Laptop 2 (LP2) as the client. As shown as fig. 2 On the server side (LP1), the system performs dataset preprocessing, model training, performance evaluation, and deployment of the prediction model. On the client side (LP2), users interact through a graphical interface to securely upload test data and obtain real-time predictions. The integration of a Flask-based API ensures smooth communication between client and server, while Redis-based authentication provides secure login and role-based access control. This architecture enables scalable, automated, and real-time intrusion detection, making it suitable for protecting sensitive IoMT infrastructures against evolving cyber threats.

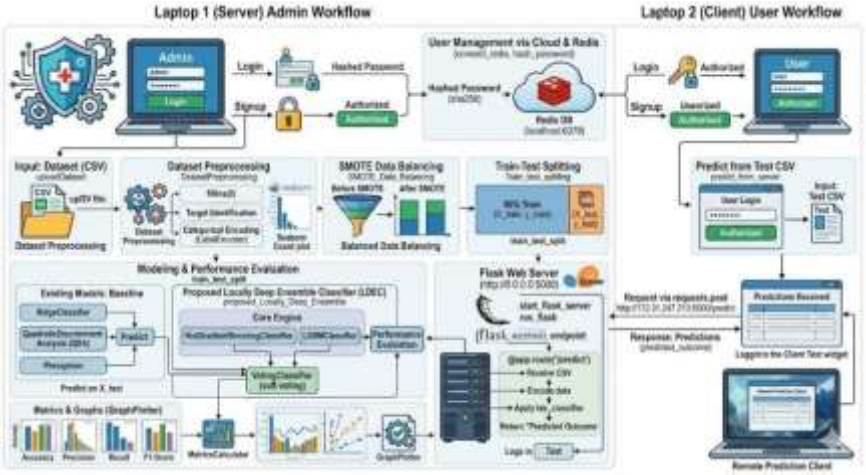


Fig. 2: Proposed System Architecture

Step 1: Setup and Library Initialization

- Essential libraries for data processing (pandas, numpy), visualization (matplotlib, seaborn), and machine learning are imported.
- Joblib is used for model persistence, and threading enables parallel execution of GUI and server.
- Redis and hashlib are used for secure authentication and password hashing.
- Custom modules such as MetricsCalculator and GraphPlotter are integrated for evaluation and visualization.

Step 2: Dataset Loading

- The admin uploads the dataset via a Tkinter GUI.
- The dataset is read from a CSV file and displayed for verification.
- This step ensures correctness before further processing.

Step 3: Data Preprocessing

- Missing values are handled and categorical features are encoded using LabelEncoder.
- The dataset is split into features (X) and target (Y).

- A count plot is generated to visualize class distribution.

Step 4: Data Balancing (SMOTE)

- SMOTE is applied to handle class imbalance.
- Synthetic samples are generated for minority classes.
- Graphs compare distributions before and after balancing.

Step 5: Train-Test Splitting

- The dataset is divided into 80% training and 20% testing data.
- This ensures proper evaluation on unseen data.

Step 6: Model Training and Evaluation

- Baseline models: Ridge Classifier, QDA, and Perceptron are trained.
- Proposed model: LDEC combines HistGradientBoosting and LightGBM using soft voting.
- Models are evaluated and stored using joblib.

Step 7: Performance Analysis

- Metrics such as Accuracy, Precision, Recall, and F1-score are calculated.
- Graphs and summaries are generated for comparison.
- The best-performing model (LDEC) is selected.

Step 8: Flask Server Deployment

- The trained LDEC model is deployed using a Flask server.
- A REST API endpoint (/predict) is created.
- The server processes uploaded data and returns predictions in JSON format.

Step 9: Authentication Using Redis

- Redis stores user credentials securely.
- Passwords are hashed using SHA-256.
- Role-based access (Admin/User) is implemented.

Step 10: Client-Side Prediction (LP2)

- Users log in via GUI and upload test CSV files.
- The file is sent to the server via HTTP POST request.
- The server processes and predicts outcomes.

Step 11: Result Display

- Predictions are displayed in the user interface.
- Server logs track incoming data and outputs.
- Suspicious activities can be identified and monitored.

Step 12: System Summary

- **LP1 (Server):** Handles preprocessing, training, evaluation, and deployment.
- **LP2 (Client):** Provides user interface for prediction.
- **Flask API:** Enables communication between client and server.
- **Redis:** Ensures secure authentication and access control.

4. RESULTS ANALYSIS

Fig. 3 illustrates the main interface that serves as the entry point for system administrators. This screen allows authorized users to access system functionalities through proper authentication options. It ensures that administrative control is granted only to verified users for maintaining system security. The interface provides a clear layout for navigation and further interaction with other modules. It acts as the initial access point to manage data operations and model analysis.

Fig. 3: Admin Screen

Fig. 4 depicts the administrator's control panel, which provides multiple analytical options for data handling and model execution. Through this dashboard, the admin can perform dataset uploading, preprocessing, balancing, and various model evaluations. It integrates essential functionalities required to run existing and proposed machine learning models. The interface promotes organized execution of tasks in a single workspace. It ensures efficient management of every operation within the IDS system.

Fig. 4: Admin Dashboard Screen

Fig. 5 shows the comparison between class distributions before and after applying the SMOTE algorithm. The left graph indicates the original imbalance where some classes have fewer samples, while the right graph displays equalized class counts post-balancing. SMOTE helps the model learn better by generating synthetic samples for minority classes. This process ensures fairness in model training and reduces bias toward majority classes. It enhances the robustness and overall predictive performance of the IDS model.

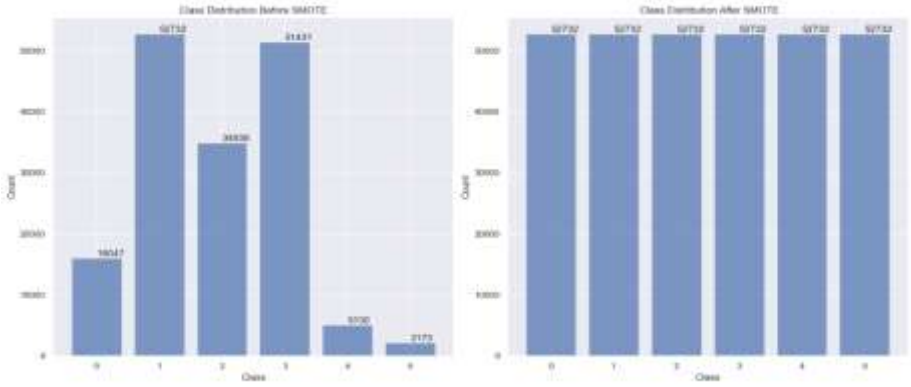
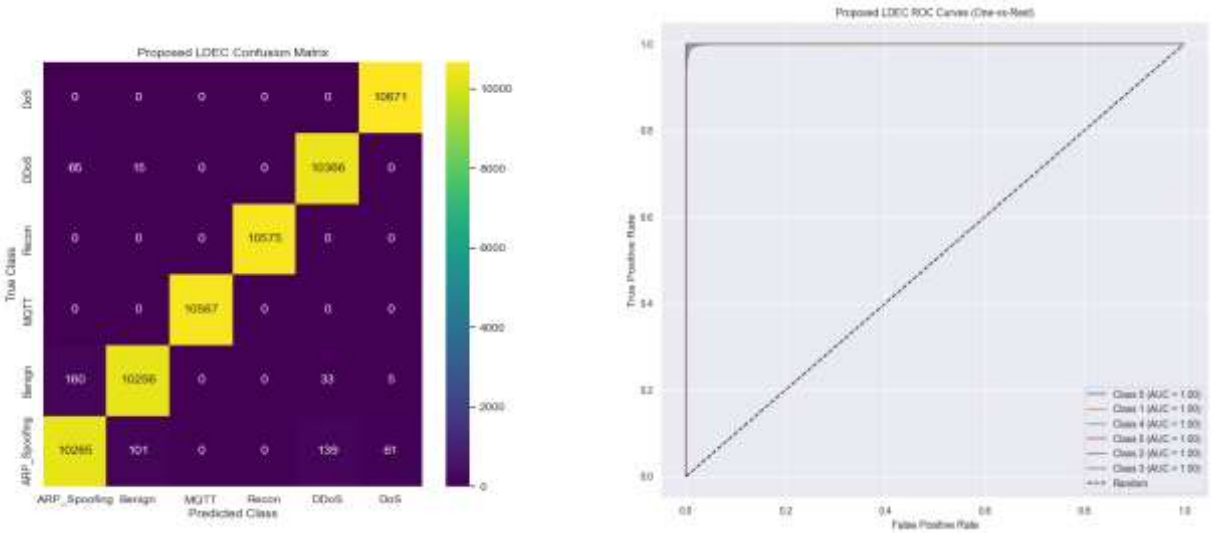


Fig. 5: Class Distribution of Dataset Before and After Applying SMOTE



(a) (b)

Fig. 6: Proposed LDEC (a) Confusion Matrix and (b) ROC Curve

Fig. 6(a) illustrates the confusion matrix for the proposed LDEC model, clearly showing dominant diagonal elements representing correctly classified samples. The minimal off-diagonal values indicate very few misclassifications across classes like ARP Spoofing, Benign, and DDoS. This signifies strong feature extraction and decision fusion achieved through local deep ensemble learning.

Fig. 6(b) presents the ROC curves for all classes, each achieving an AUC of 1.00, reflecting perfect class separability and predictive reliability. The model demonstrates consistent performance across every intrusion category, outperforming all traditional classifiers and proving its effectiveness for real-time IoMT intrusion detection.

Fig. 7 depicts the interface showing the initiation of the local Flask server, which enables web-based interaction for real-time prediction and monitoring. Once started, the server runs at a specified local address and port, allowing users to access the IDS functionalities through the browser. This integration bridges the trained LDEC model with a deployable interface for live threat detection. The screen confirms the system's operational readiness for secure communication and intelligent intrusion prevention in IoMT architectures.

Fig. 7: Start Server Screen

Fig. 8 illustrates the interface where users can upload a test CSV file to obtain predictions from the trained model. The uploaded dataset is analyzed in real time, and the output is generated through the proposed deep ensemble model. This feature enables seamless integration of real-world IoMT data into the prediction workflow. It empowers users to perform intrusion analysis without direct technical intervention. The design supports automation and ensures quick data-driven outcomes.

Fig. 8: Predict from Test CSV (Laptop 2) Screen

Fig. 9 shows the admin-side interface that receives the dataset transmitted by the client system. The server decodes the incoming data, processes it through the trained model, and determines the corresponding intrusion category. This setup demonstrates effective cross-device communication in a

distributed IoMT environment. It validates the integration of client data with server-side computation. The screen confirms successful data reception and real-time processing.

Fig. 9: Cloud Predicting the Test Data of the Client & Transmitting Screen

Fig. 10 represents the user-side interface that receives the prediction results from the admin server. Once the data is processed and classified, the predicted outcomes are transmitted back to the client device. The results indicate detected intrusion types such as DoS, DDoS, or ARP Spoofing. This communication ensures timely alerts and intelligent decision support for users. The setup demonstrates a complete client-server interaction flow, ensuring real-time detection and secure feedback within the IoMT ecosystem.

Fig. 10: User Receiving the Predictions Screen

9.3 Comparative Analysis

The performance comparison table 1 shows a clear improvement of the proposed LDEC model over existing approaches. The Perceptron model records low performance with 31.03% accuracy and weak precision, recall, and F1-score values, indicating poor classification capability. QDA improves the results with 69.77% accuracy and relatively better precision (79.53%), but still lacks balanced performance across metrics. Ridge further enhances performance, achieving 82.62% accuracy along with stable precision, recall, and F1-score values. However, the proposed LDEC model significantly outperforms all existing models with an accuracy of 99.08%. It also achieves extremely high precision (99.83%), recall (99.03%), and F1-score (99.08%), indicating near-perfect classification. The results highlight the strong generalization and robustness of the proposed approach.

Table 1: Comparative Analysis of the Classification Models

Algorithm	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
Existing Perceptron	31.0387	21.8942	31.0856	21.5633
Existing QDA	69.7799	79.5331	69.7623	66.2670
Existing Ridge	82.6214	83.0509	82.6124	81.9450
Proposed LDEC	99.0850	99.8316	99.0319	99.0805

5. CONCLUSION

In this research User Cloud Driven IDS for Securing IoMT provides an intelligent, scalable, and efficient solution for enhancing the cybersecurity of connected healthcare infrastructures. By integrating cloud-based analytics with machine learning algorithms such as Histogram Gradient Boosting Classifier and Light Gradient Boosting Machine Classifier within a Voting Ensemble, the system achieves superior accuracy and robustness compared to traditional models like Ridge, QDA, and Perceptron. Through effective preprocessing, normalization, and SMOTE-based data balancing, it successfully mitigates data imbalance issues and enhances model generalization across diverse IoMT traffic patterns. The hybrid deployment using Tkinter GUI, Redis-based authentication, and a Flask server ensures real-time user interaction and secure remote predictions. Experimental results demonstrate significant improvements in detection accuracy, recall, and precision, confirming the reliability of the Locally Deep Ensemble Classifier (LDEC). The system contributes a powerful and adaptive intrusion detection framework capable of protecting IoMT environments against evolving cyber threats while maintaining high performance and scalability.

REFERENCES

[1] Halder, S.; Ghosal, A.; Conti, M. Efficient physical intrusion detection in Internet of Things: A Node deployment approach. Comput. Netw. 2019, 154, 28–46.

[2] Kumar, P.; Gupta, G.P.; Tripathi, R. An ensemble learning and fog-cloud architecture-driven cyber-attack detection framework for IoMT networks. Comput. Commun. 2021, 166, 110–124.

[3] Balandina, E.; Balandin, S.; Koucheryavy, Y.; Mouromtsev, D. IoT use cases in healthcare and tourism. In Proceedings of the 2015 IEEE 17th Conference on Business Informatics, Lisbon, Portugal, 13-16 July 2015; IEEE: Piscataway, NJ, USA, 2015; Volume 2, pp. 37–44.

[4] A. Heidari, M.A.J. Jamali Internet of Things intrusion detection systems: a comprehensive review and future directions.Clust. Comput., 26 (2023), pp. 3753-3780, 10.1007/s10586-022-03776-z

[5] Alalhareth, K; Hong, J.C. An Improved Mutual Information Feature Selection Technique for Intrusion Detection Systems in the Internet of Medical Things. Sensors 2022, 21, 4275.

[6] Alalhareth, M.; Hong, S.-C. An Adaptive Intrusion Detection System in the Internet of Medical Things Using Fuzzy-Based Learning. Sensors 2023, 23, 9247. https://doi.org/10.3390/s23229247

[7] Salehpour, A., Balafar, M.A. & Souri, A. An optimized intrusion detection system for resource-constrained IoMT environments: enhancing security through efficient feature selection and classification. J Supercomput 81, 783 (2025). https://doi.org/10.1007/s11227-025-07253-3

[8] Sharma, N., Shambharkar, P.G. Multi-layered security architecture for IoMT systems: integrating dynamic key management, decentralized storage, and dependable intrusion detection framework. Int. J. Mach. Learn. & Cyber. 16, 6399–6446 (2025). https://doi.org/10.1007/s13042-025-02628-7

[9] Zachos, G.; Mantas, G.; Porfyrakis, K.; Rodriguez, J. Implementing Anomaly-Based Intrusion Detection for Resource-Constrained Devices in IoMT Networks. Sensors 2025, 25, 1216. https://doi.org/10.3390/s25041216

[10] Hafid, A.; Rahouti, M.; Aledhari, M. Optimizing Intrusion Detection in IoMT Networks Through Interpretable and Cost-Aware Machine Learning. Mathematics 2025, 13, 1574. https://doi.org/10.3390/math13101574

[11] Alserhani, F. Intrusion Detection and Real-Time Adaptive Security in Medical IoT Using a Cyber-Physical System Design. Sensors 2025, 25, 4720. https://doi.org/10.3390/s25154720

[12] Khan, M.Z.; Sabur, A.; Ghandorh, H. A Novel Internet of Medical Things Hybrid Model for Cybersecurity Anomaly Detection. Sensors 2025, 25, 6501. https://doi.org/10.3390/s25206501

[13] Naghib, A., Gharehchopogh, F.S. & Zamanifar, A. A comprehensive and systematic literature review on intrusion detection systems in the internet of medical things: current status, challenges, and opportunities. Artif Intell Rev 58, 114 (2025). https://doi.org/10.1007/s10462-024-11101-w

[14] Y. Rbah, M. Mahfoudi, M. Fattah, Y. Balboul, S. Mazer and M. Elbekkali, "An Intrusion Detection System For Internet of Medical Things Using Machine Learning Approaches," 2025 5th International Conference on Innovative Research in Applied Science, Engineering and Technology (IRASET), Fez, Morocco, 2025, pp. 1-6, doi:10.1109/IRASET64571.2025.11008243.

[15] Jordi Doménech, Olga León, Muhammad Shuaib Siddiqui, Josep Pegueroles,Evaluating and enhancing intrusion detection systems in IoMT: The importance of domain-specific datasets,Internet of Things,Volume 32,2025,101631, ISSN 2542-6605 https://doi.org/10.1016/j.iot.2025.101631.