



International Journal of Artificial Intelligence and Machine Learning

Publisher's Home Page: <https://www.svedbergopen.com/>



Research Paper

Open Access

Edge-Deployed Clinical Monitoring and Intrusion Detection Secured by Hyperledger Blockchain Decentralized Storage in Medical IoT

Reema Shaheen^{1*}, Adnan Shahid Khan¹, Asiya Aman², Salman Nusrat² and Muhammad Shahid Dildar^{1,3}

¹Faculty of Computer Science and Information Technology, Universiti Malaysia Sarawak, Kota Samarahan, 94300, Malaysia.

²Faculty of Business and Management, Muscat University, Oman, 130, Saudi Arabia

³Department of Business Informatics, King Khalid University, Abha, Saudi Arabia

Corresponding mail: shaheen5392@yahoo.com

Abstract

Medical Internet of Things (MIoT) devices are vital to the functioning of intensive care units, allowing patient vital signs to be transmitted in real time but they are also a double-edged sword: network-level attacks can lead to the integrity of the data being compromised and delayed or failed clinical action can occur if a critical condition is not detected. This paper addresses both risks with a single edge-deployed framework called BlockEdge-Health. A Histogram Gradient Boosting (HGB) classifier trained on MIMIC-IV vital signs identifies critical patient states at the edge, while a K-Nearest Neighbors (KNN) intrusion detector trained on NSL-KDD guards the network layer. Every validated decision is committed to a Hyperledger Fabric permissioned ledger through a smart contract, and bulk medical files are stored on IPFS with cryptographic hash anchors held off-chain. On MIMIC-IV, the clinical module achieved 97.8% test accuracy and a cross-validated ROC-AUC of 0.998. The KNN detector reached 97.6% accuracy across ten cross-validation folds (97.82% precision, 97.97% recall, 97.89% F1, all with standard deviations below 1%). The Fabric network ran at 98 transactions per second under baseline load and around 145 TPS at steady state, with average latency dropping from 0.48 s during initialization to 0.31 s after peer synchronization. IPFS hash verification held at 100% accuracy, and storage overhead was 12.8 MB per 1,000 transactions, which is substantially less than the 18 to 20 MB reported by Ethereum-based comparison systems. The results show that a carefully designed edge architecture can serve both the clinical and cybersecurity demands of an ICU without cloud dependence and without trading one property against the other.

Keywords: MIoT; Edge Computing; Clinical Decision Support; Intrusion Detection; Histogram Gradient Boosting; K-Nearest Neighbors; Hyperledger Fabric; IPFS; Permissioned Blockchain, and Healthcare Security.

1. INTRODUCTION

In an intensive care unit, a large amount of physiological data is constantly flowing. Every single bed has ECG leads, pulse oximeters, blood pressure cuffs, and infusion pumps, all connected to a single hospital network, which in turn drives clinical decisions in seconds instead of minutes. IoT sensor networks have also been applied for continuous and real-time health-condition monitoring, demonstrating the potential of connected sensing environments for timely health assessment (Li et al., 2026) [53]. The long-term consequences of delays in recognizing a deteriorating patient state are confirmed by large retrospective analyses of the records from the ICU. The implication is clear: the monitoring system must also continue to operate, and the information that it transports must be reliable. Both requirements are harder than they look. Hospital networks face documented attack patterns that include denial-of-service floods aimed at monitoring gateways, man-in-the-middle injections that alter vital-sign readings in transit, replay attacks, and unauthorized probing of patient record stores. Machine-learning security research applied specifically to healthcare has shown that commercially deployed medical devices are susceptible to adversarial manipulation under standard network conditions. Aldmour's performance

analysis of Blockchain of Things systems catalogues similar attack vectors in networked ICU settings. The attack surface exists and is well characterized; the question is how to defend it without adding overhead that slows down the clinical data path.

Most hospitals still send monitoring data to central cloud servers. That arrangement works under normal load, but three failure modes appear under pressure. Network latency adds delay to the alert pipeline, and in an ICU where clinical windows are short, that delay has a cost. A cloud link that goes down, even for minutes, silences every alert that depends on it. And each byte of raw physiological data leaving the ward and travelling to a remote server traverses a path that could be intercepted. Shi et al. argued that latency-sensitive healthcare applications are one of the primary cases for moving compute closer to the data source, and the failure modes listed above are exactly what motivates that shift. Edge computing reduces the first two problems by running inference at a local gateway node so that alert generation continues regardless of what the external network is doing. Satyanarayanan traced the broader emergence of edge deployments and observed that healthcare and autonomous systems drove adoption faster than other sectors because their tolerance for cloud-round-trip latency is effectively zero. The remaining problem is governance: an edge node without external accountability can quietly alter what it stores. Nothing in a standard edge deployment prevents a compromised gateway from modifying its local logs after the fact.

Blockchain is the natural answer to the governance problem, but the choice of platform matters. Public chains charge per-transaction fees, expose participation to any observer, and use proof-of-work consensus that is too slow for continuous ICU event logging. Dagher et al. showed that a consortium chain with role-based smart contracts can enforce privacy-preserving access control for electronic health records without any off-chain policy engine. Azaria et al. demonstrated the principle earlier with MedRec on Ethereum but noted that public-chain economics make the design impractical at ICU data rates. Hyperledger Fabric avoids those limits: only authenticated actors can transact, smart contracts encode access policy directly in the network logic, and the Raft ordering service provides fast finality without mining. Androulaki et al. described the Fabric architecture in detail and showed that its pluggable consensus and channel-based privacy model give it measurable advantages over public chains for enterprise deployments. The storage problem is separate. An ICU produces waveforms, monitoring logs, and nursing notes that are far too large to write to any ledger. Storing them on-chain inflates block size, slows propagation, and raises storage costs at every peer. Benet's InterPlanetary File System addresses this by replacing file locations with content identifiers derived from the file's own hash: the file lives off-chain in a distributed network, and the ledger holds only the 32-byte SHA-256 identifier. Any modification to the stored file produces a different identifier, so a mismatch at retrieval is immediately detected. Mishra et al. measured this pattern in an IoT healthcare context and found 32% lower storage overhead than full on-chain equivalents with no loss of integrity assurance. This paper brings the four layers together in a framework called BlockEdge-Health. Two machine-learning modules run on the same edge node. The first classifies patient vital signs from the MIMIC-IV clinical dataset using Histogram Gradient Boosting, selected after comparing it with Logistic Regression, Random Forest, XGBoost, and a soft-voting ensemble. Unsupervised anomaly labels from an Isolation Forest serve as the supervision target for the classifier since manually labelled deterioration events are not available for all records in MIMIC-IV. The second module screens network traffic with a K-Nearest Neighbors detector trained on NSL-KDD, using SMOTEENN applied inside each cross-validation fold to handle class imbalance without leaking synthetic samples across folds. All validated decisions go to Hyperledger Fabric through a smart contract. All large files go to IPFS with their cryptographic hashes anchored on the ledger. A doctor-facing interface delivers real-time alerts and answers blockchain-authenticated queries over historical records.

The paper is organized as follows. Section II reviews related work. Section III describes the proposed framework and its algorithms. Section IV gives the experimental setup. Section V reports the results. Section VI discusses them. Section VII highlights the limitations and future work of the research, and Section VIII concludes the paper.

2. LITERATURE REVIEW

Recent advances in secure healthcare and Medical IoT (MIoT) have increasingly combined artificial intelligence, edge computing, blockchain, decentralized storage, and intrusion detection to address the security, privacy, scalability, and real-time monitoring requirements of connected medical environments. Siqueira et al. presented a comprehensive survey of intrusion detection systems for Healthcare 5.0, highlighting emerging concepts, challenges, and practical applications for securing intelligent healthcare infrastructures (Siqueira et al., 2025) [1]. Al-Otaibi et al. proposed an AI-driven security framework for medical sensor networks that emphasizes privacy and trust in smart healthcare systems (Al-Otaibi et al., 2025) [2], while Rajpurkar et al. reviewed the broader role of artificial intelligence in health and medicine, establishing the importance of AI-assisted clinical decision-making and healthcare applications (Rajpurkar et al., 2022) [3]. Recent studies have further highlighted the application of artificial intelligence and machine-learning approaches in healthcare-related prediction and decision-support systems (Quazi et al., 2022) [51]. For IoT security, Al-Garadi et al. surveyed machine learning and deep learning

approaches for detecting and mitigating security threats in IoT environments (Al-Garadi et al., 2020) [4], and Newaz et al. introduced HealthGuard, a machine-learning-based security framework specifically designed for smart healthcare systems (Newaz et al., 2019) [5]. Aldmour investigated Blockchain of Things (BCoT) systems with particular emphasis on scalability and efficiency, providing a foundation for integrating blockchain with resource-constrained IoT environments (Aldmour, 2024) [6]. Maftai et al. proposed a blockchain framework for scalable, high-density IoT networks, demonstrating the relevance of blockchain-based architectures for large-scale connected environments (Maftai et al., 2025) [7]. Edge computing provides another important layer for healthcare monitoring because it enables computation closer to data-generating devices; Shi et al. discussed the vision and challenges of edge computing for distributed IoT environments (Shi et al., 2016) [8], while Satyanarayanan described the emergence of edge computing as a paradigm for reducing latency and supporting geographically distributed applications (Satyanarayanan, 2017) [9]. In healthcare data protection, Bezanjani et al. investigated a synergistic approach combining deep learning and blockchain for privacy-preserving healthcare data in IoT environments (Bezanjani et al., 2025) [10]. Blockchain-based medical data management was further explored by Azaria et al. through MedRec, which uses blockchain for medical data access and permission management (Azaria et al., 2016) [11], and by Dagher et al. through Ancile, a privacy-preserving framework for access control and interoperability of electronic health records using blockchain technology (Dagher et al., 2018) [12]. Kokate and Shrawankar addressed secure IoT data transmission in healthcare using blockchain and a trust-weighted Proof-of-Authority consensus mechanism, demonstrating the applicability of blockchain consensus to healthcare communication security (Kokate & Shrawankar, 2025) [13]. A key technological foundation for permissioned healthcare blockchain networks was provided by Androulaki et al., who introduced Hyperledger Fabric as a distributed operating system for permissioned blockchains (Androulaki et al., 2018) [14]. Mani et al. extended this concept to healthcare through Hyperledger HealthChain, combining Hyperledger with IPFS-based patient-centric storage for health records (Mani et al., 2021) [15]. Similarly, Jayabalan and Jeyanthi developed a scalable blockchain architecture using off-chain IPFS storage to enhance healthcare data security and privacy (Jayabalan & Jeyanthi, 2022) [16], while Benet introduced IPFS as a content-addressed, versioned peer-to-peer file system suitable for decentralized data management (Benet, 2014) [17]. Mishra et al. further investigated the integration of blockchain and IPFS for healthcare data management and sharing within IoT environments, reinforcing the potential of decentralized storage for connected healthcare systems (Mishra et al., 2025) [18]. Karadaş and Kalkan proposed IBAM, an IPFS- and blockchain-based authentication mechanism for the MQTT protocol in IoT, demonstrating how decentralized technologies can strengthen authentication and communication security in IoT networks (Karadaş & Kalkan, 2023) [19]. From the clinical-data perspective, Johnson et al. introduced MIMIC-IV, a freely accessible electronic health record dataset that supports research involving clinical monitoring and intelligent healthcare analytics (Johnson et al., 2023) [20]. Machine-learning methods can complement blockchain and edge security mechanisms by enabling automated detection and classification; Chen and Guestrin introduced XGBoost as a scalable tree-boosting system for predictive modeling (Chen & Guestrin, 2016) [21], while Liu et al. proposed Isolation Forest for efficient anomaly detection (Liu et al., 2008) [22]. Tavallaee et al. provided a detailed analysis of the KDD Cup 99 dataset, which has been widely used in the development and evaluation of intrusion detection approaches (Tavallaee et al., 2009) [23], and Chawla et al. introduced SMOTE to address class imbalance through synthetic minority over-sampling, an important consideration for security datasets containing comparatively fewer attack samples (Chawla et al., 2002) [24]. In clinical monitoring, Reyna et al. demonstrated the importance of early prediction of sepsis from clinical data through the PhysioNet/Computing in Cardiology Challenge 2019 (Reyna et al., 2020) [25]. Kaissis et al. examined secure, privacy-preserving, and federated machine learning for medical imaging, emphasizing the importance of protecting sensitive healthcare information during collaborative AI development (Kaissis et al., 2020) [26]. Ring et al. surveyed network-based intrusion detection datasets and discussed their characteristics for evaluating intrusion detection systems (Ring et al., 2019) [27], while Buczak and Guven reviewed data mining and machine-learning methods for cybersecurity intrusion detection (Buczak & Guven, 2015) [28]. Moustafa et al. provided a comprehensive review of network anomaly detection systems, further establishing anomaly identification as a major component of network security (Moustafa et al., 2019) [29]. For IoT-specific threat detection, Diro and Chilamkurti developed a distributed attack detection scheme using deep learning for IoT environments (Diro & Chilamkurti, 2018) [30]. Blockchain programmability and consensus mechanisms are also important to decentralized healthcare architectures; Zheng et al. surveyed smart contracts, their challenges, advances, and platforms (Zheng et al., 2020) [31], while Hasselgren et al. reviewed blockchain applications across healthcare and health sciences, identifying opportunities for secure and decentralized healthcare data management (Hasselgren et al., 2020) [32]. Xiao et al. surveyed distributed consensus protocols for blockchain networks, providing a broader foundation for understanding consensus in decentralized systems (Xiao et al., 2020) [33], and Li et al. investigated blockchain for large-scale IoT data storage and protection, directly connecting blockchain with secure management of distributed IoT data (Li et al., 2018) [34]. Rieke et al. examined federated learning for digital health, emphasizing distributed learning approaches that can reduce the need to centralize sensitive healthcare data (Rieke et al., 2020) [35]. Within

healthcare blockchain cybersecurity, Selvarajan and Mouratidis proposed a quantum trust and consultative transaction-based blockchain cybersecurity model for healthcare systems, strengthening trust-oriented blockchain protection mechanisms (Selvarajan & Mouratidis, 2023) [36]. Selvarajan et al. developed a lightweight AI-based blockchain security model addressing security and privacy in Industrial IoT systems, demonstrating the integration of intelligent analysis with blockchain-based protection (Selvarajan et al., 2023) [37]. Shitharth et al. investigated edge computing combined with IoT and incremental learning, highlighting the potential of edge intelligence for distributed real-time processing (Shitharth et al., 2023) [38], while Shitharth et al. proposed an intelligent multi-attack intrusion detection algorithm for edge-assisted IoT, directly supporting the integration of intrusion detection with edge-based architectures (Shitharth et al., 2023) [39]. For medical IoT applications, Khadidos et al. introduced the TasLA deep convolution-based data-fusion model for IoMT smart healthcare, demonstrating intelligent processing of heterogeneous healthcare data (Khadidos et al., 2023) [40]. Rabie et al. developed a privacy-preserving distributed certificate-less aggregate signature authentication scheme for healthcare wearable wireless medical sensor networks, addressing secure authentication and privacy protection for medical sensing environments (Rabie et al., 2023) [41]. Aluvalu et al. investigated privacy-enhanced blockchain for efficient data transmission over wireless communication, demonstrating the potential of blockchain to protect distributed data during transmission (Aluvalu et al., 2023) [42]. Manoharan et al. examined IoT implementation with blockchain and machine learning for enhanced security, further connecting intelligent detection with blockchain-based protection (Manoharan et al., 2023) [43]. More recent work has also demonstrated the combined use of IoT sensing, artificial intelligence, and blockchain to support secure data collection, intelligent analysis, anomaly identification, and decentralized data governance (Patkar et al., 2026) [52]. Nandagopal et al. proposed a Deep Auto-Optimized Collaborative Learning model for disease prognosis using AI-IoMT systems, illustrating the role of collaborative intelligence in medical IoT applications (Nandagopal et al., 2024) [44]. Maruthupandi et al. developed an intelligent attention-based deep convoluted learning model for smart healthcare security, emphasizing deep-learning-driven protection of healthcare systems (Maruthupandi et al., 2025) [45]. Kumar et al. proposed a lightweight and secure authentication and privacy-protection scheme for IoMT, addressing the security requirements of resource-constrained medical IoT environments (Kumar et al., 2025) [46]. Kailasanathan et al. investigated an interpretable framework for heart disease prediction using IoMT sensors, demonstrating the role of sensor-driven clinical analytics in intelligent healthcare monitoring (Kailasanathan et al., 2025) [47]. Natarajan et al. proposed a quantum-secure patient login credential system using blockchain for electronic health record sharing, strengthening blockchain-based identity and access protection in healthcare (Natarajan et al., 2025) [48]. Selvarajan et al. investigated IoT-driven healthcare monitoring using evolutionary optimization and game theory, highlighting intelligent optimization for healthcare monitoring environments (Selvarajan et al., 2025) [49]. Finally, Pramanick et al. investigated stacking machine-learning models and optimization for improved cyberattack detection, demonstrating the potential of ensemble learning and optimization for identifying malicious activities (Pramanick et al., 2025) [50]. Collectively, these studies establish the major technological foundations for secure edge-deployed clinical monitoring, including edge-based low-latency processing [8], [9], [38], [39], IoMT-based clinical sensing and intelligent healthcare analytics [40], [44], [47], [49], AI-based intrusion and cyberattack detection [1], [4], [5], [21], [22], [28], [29], [30], [39], [50], blockchain-based security, authentication, access control, and trust [10]–[16], [31]–[34], [36], [37], [42], [43], [48], and decentralized IPFS-based healthcare data storage [53–55]. These works collectively provide the technological basis for integrating edge clinical monitoring with intrusion detection and permissioned blockchain-based decentralized storage, while Hyperledger Fabric [14] and Hyperledger HealthChain [15] provide particularly relevant foundations for a Hyperledger-oriented healthcare architecture.

3. PROPOSED MODEL

BlockEdge-Health has four cooperating layers. Bedside MIoT devices send raw vitals and network traffic to a local edge node. The edge node runs two AI modules and packages each decision as a transaction for Hyperledger Fabric. The Fabric layer endorses, orders, and commits the transaction. Large files go to IPFS, with their content identifiers anchored on the ledger. A doctor interface sits above all four layers, receiving real-time alerts and answering role-controlled queries. Figure 1 shows the full proposed system architecture.

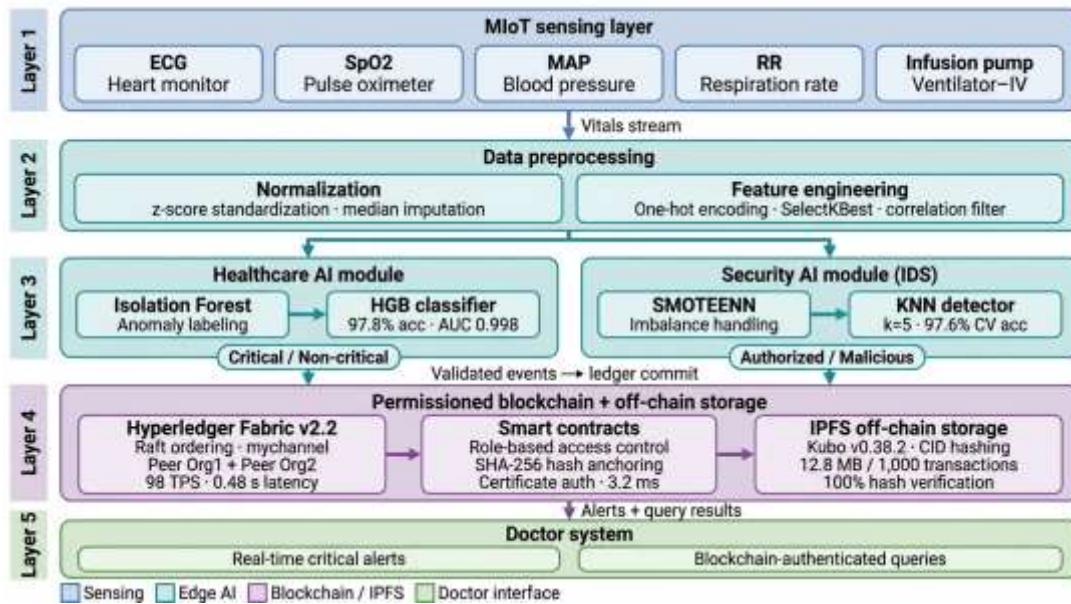


Figure 1. BlockEdge-Health System Architecture.

3.1 Preprocessing Pipeline

The raw sensor data comes from five channels (heart rate (HR), respiratory rate (RR), oxygen saturation (SpO₂), mean arterial pressure (MAP), and network flow features) and is then combined into a shared preprocessing stage before being fed into either AI module. Missing values in continuous channels are filled by linear interpolation; discrete fields use median substitution. Every remaining feature is standardized:

$$x_i' = \frac{x_i - \mu_i}{\sigma_i} \quad (1)$$

where μ and σ are computed on the training portion and fixed for inference. Features with a Pearson correlation below 0.15 with the target are dropped. This step preserves about a third of the original NSL-KDD columns without one-hot encoding and avoids having the distance computation in KNN dominated by uninformative attributes.

3.2 Healthcare AI Module

The clinical module takes in the four important features of vital signs and determines if the patient's condition is critical or not. The magnitude of the anomalous readings is not available for all the MIMIC-IV records, so the first step is to label the anomalous readings as the positive class with the help of an Isolation Forest. The item becomes the item that is being supervised. The classifiers used for model selection were: Logistic Regression, Random Forest, XGBoost, a soft voting ensemble of all four classifiers, and Histogram Gradient Boosting. HGB won on every cross-validated metric.

HGB was trained with an 80/20 stratified split, stratified five-fold cross-validation, a learning rate of 0.1, up to 500 boosting iterations, and early stopping triggered after 30 rounds without improvement. Algorithm 1 gives the full procedure.

Algorithm 1: HGB Clinical Classifier (MIMIC-IV)

Input : MIMIC-IV dataset; features {HR, RR, SpO₂, MAP}
 anomaly labels from IsolationForest
 Output: class y' in {0=normal, 1=critical}; ledger hash H
 impute nulls; z-score each feature (Eq. 1)
 drop features with $|r| < 0.15$
 labels $\leftarrow$ IsolationForest(contamination=auto)
 HGB $\leftarrow$ HistGradientBoosting(lr=0.1, iter=500, stop=30)
 stratified 80/20 split; fit on training set
 evaluate: stratified 5-fold CV (F1, recall, ROC-AUC)
 for each live input x_{live} :
 $y' \leftarrow$ HGB.predict(x_{live})
 if $y' == 1$: fire_alert(doctor_system)

```
H <- SHA-256(y' || timestamp)
submit_to_fabric(y', H, role='healthcare')
return y', H
```

After validation, the model is quantized for the edge node. Floating-point weights are replaced by 8-bit integers:

$$Q(x) = \text{round}\left(\frac{X}{S}\right) + Z \quad (2)$$

S is the per-layer scale factor, and Z is a zero-point offset. Quantization still maintains the same accuracy that would be obtained on the full-precision model but uses less memory than a Raspberry Pi-type device can handle

3.3 Intrusion Detection Module

The security module monitors network traffic on the NSL-KDD benchmark. There are 125,973 training records and 22,544 test records, where each of these records has 41 raw features that would amount to 122 columns after applying label and one-hot encoding to the three categorical columns (protocol type, service, and flag). No synthetic samples cross fold boundaries due to class imbalance, which is addressed with SMOTEENN within each fold of the cross-validation.

1) Distance metric: K-Nearest Neighbors with $k = 5$, distance is measured by Euclidean distance, and the result is classified by distance:

$$d(x, x_j) = \sqrt{\sum_k (x_k - x_{jk})^2} \quad (3)$$

The label predicted is the distance-weighted majority voting of the 5 nearest neighbors:

$$y' = \text{mode}\{y_j : x_j \in N_5(x)\} \quad (4)$$

Algorithm 2 gives the full pipeline.

Algorithm 2: KNN Intrusion Detector (NSL-KDD)

Required	Input: NSL-KDD CSVs; $k=5$; metric=Euclidean
	Output: labels y_{pred} ; confusion matrix cm ; metrics
1	load 125973-train / 22544-test; assign 42 column names
2	one-hot encode {protocol_type, service, flag} -> 122 feats
3	map attack labels -> 1, normal -> 0 (binary target)
4	StandardScaler.fit(X_{train}); .transform(X_{train} , X_{test})
5	inside each CV fold: SMOTEENN.fit_resample(X , y)
6	knn <- KNN($n_{\text{neighbors}}=5$, weights='distance')
7	knn.fit($X_{\text{train_resampled}}$, $y_{\text{train_resampled}}$)
8	evaluate via 10-fold stratified CV
9	y_{pred} <- knn.predict(X_{test}) [raw partition]
10	cm <- confusion_matrix(y_{test} , y_{pred})
11	for each $y_{\text{pred}}==1$: submit_to_fabric(event, role='security')
12	return y_{pred} , cm , metrics

3.4 Permissioned Blockchain Layer

The ledger uses Hyperledger Fabric v2.2 with the Raft ordering service, one orderer node, two peer organizations (Org1: hospital, Org2: cloud service), and a certificate authority that manages digital identities. There is only one channel called mychannel through which all transactions are processed. Smart contracts in JavaScript check the transaction payload, adhere to role-based access, and store a hash of the event content (SHA-256):

$$H(E) = \text{SHA} - 256(E \parallel \text{timestamp}) \quad (5)$$

The hash is the on-chain footprint of every AI decision. SHA-256 is collision-proof, so if a malicious person changes an event after it has been committed, it will result in a mismatch in hash codes and will be detected by the next query. The health of the network is available in real time via Prometheus metrics endpoints on every peer (:9443), which include block-commit, chaincode duration, CPU-rate, and transaction-count metrics, avoiding any interaction with the consensus path.

3.5 IPFS off-chain Storage

The full records, ECG waveforms, multi-hour monitoring logs, and blood-pressure archives are uploaded to an IPFS

node using ipfs add and return a content identifier (CID) based on the hash of the file. The CID and its associated SHA-256 hash are recorded on the Fabric ledger. The actual file is only stored in IPFS. When retrieved, the system reads the file, recalculates its hash, and compares it to the hash on-chain:

$$\text{VERIFY: sha256}(\text{file}_{\text{retrieved}}) == H_{\text{stored}} \text{ on ledger} \quad (6)$$

A mismatch triggers an access denial and an audit alert. Critical records, those flagged by the Healthcare AI module as urgent, carry both the patient file and the alert metadata when written to IPFS. Non-critical records carry only the file. In both cases, the on-chain hash is the only link between the ledger and the file, keeping blocks small and block propagation fast.

3.6 Doctor interface and latency bound

The clinician layer has two modes. In alert mode, it pushes critical classifications to hospital displays and paging systems the moment the HGB module fires. In query mode, it accepts structured requests, say all SpO₂ readings below 90% for a patient over the last 24 hours, verifies the caller's role through a smart contract, fetches the IPFS records, checks their hashes against the ledger, and returns the verified data. The total edge decision latency is:

$$\Delta_t = \max(\Delta t_{\text{clinical}}, \Delta t_{\text{IDS}}) \quad (7)$$

Both modules run on independent threads, so neither stalls the other. A spike in IDS computation caused by a surge of suspicious traffic does not delay the clinical alert path.

4. EXPERIMENTAL SETUP

4.1 Hardware and Software

All the experiments were performed on an Apple workstation with an M4 processor, 16 GB of RAM, and 512 GB of SSD. One orderer, two peer containers, and a CA container were hosted on the same machine, all sharing the same Docker network as Hyperledger Fabric. The training and quantization of the AI modules were performed on the workstation, and the benchmarking of edge-mode inference was conducted by simulating Raspberry Pi throughput on the constrained CPU threads. Caliper also ran five rounds of workload, reaching 20-150 TPS, while Prometheus simultaneously recorded metrics on peers. The complete software stack is specified using Table I.

Table I. Implementation Software Stack.

Tool/framework	Version	Purpose
Python	3.10	AI implementation, preprocessing
scikit-learn / HGB	1.3+	Clinical classification, IDS
Node.js	18.20.4	Caliper runtime
Docker / Docker-Compose	27.1 / 2.24	Peer and orderer containers
Hyperledger Fabric	v2.2 (LTS)	Permissioned ledger
Hyperledger Caliper	v0.5.0	Blockchain benchmarking
IPFS (Kubo)	v0.38.2	Off-chain storage
Prometheus	v2.53.1	Peer monitoring
Grafana	v10.4	Visualization (internal)

4.2 Dataset

Two benchmark datasets were used. MIMIC-IV supplies de-identified ICU records from the Massachusetts Institute of Technology. This study used an 80/20 stratified split, z-score normalization, Isolation Forest anomaly labeling, and extracted heart rate, respiratory rate, oxygen saturation, and mean arterial pressure. NSL-KDD is a cleaned version of the KDD Cup 1999 benchmark that consists of 125,973 training records, 22,544 test records, and 41 features, which were increased to 122 after categorical encoding, at the University of New Brunswick. Both are freely available and widely used in their respective communities, thus allowing direct comparison with previous work.

4.3 Evaluation Metrics

The accuracy, precision, recall, and F1 metrics for the AI modules are used for scoring. The most significant factor in both domains is recall: Neither a missed critical patient state nor an intrusion is more expensive than a false alarm.

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \quad (8)$$

$$F1 = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (9)$$

ROC-AUC reports the classifier's discriminative ability independent of threshold. The blockchain is scored by per-round average latency, committed TPS, block confirmation time, CPU and memory use, and transaction success rate. IPFS is scored by storage overhead per 1,000 transactions and hash-verification accuracy.

4.4 Healthcare AI

On the MIMIC-IV test split, the best results, across all measures, were achieved by HGB, with an average prediction latency of 180 ms (Table II, Figure 2). The difference between HGB and the second-best model, the voting ensemble with a 97.2% accuracy, is 0.6 percentage points on accuracy; more significantly, it is 10 milliseconds faster per prediction; that's the speed of an alert path. Logistic Regression was the second most erratic, showing that the relationships between the vital signs here are not linear and that a linear classifier is not the right approach.

Table II. Healthcare Classifier Comparison (MIMIC-IV Test Split).

Model	Accuracy (%)	Precision	Recall	F1	Latency (ms)
Logistic Regression	92.4	0.90	0.88	0.89	220
Random Forest	95.1	0.94	0.93	0.93	210
XGBoost	96.8	0.96	0.94	0.95	200
Voting Ensemble	97.2	0.96	0.95	0.96	190
HGB (Proposed)	97.8	0.97	0.96	0.97	180

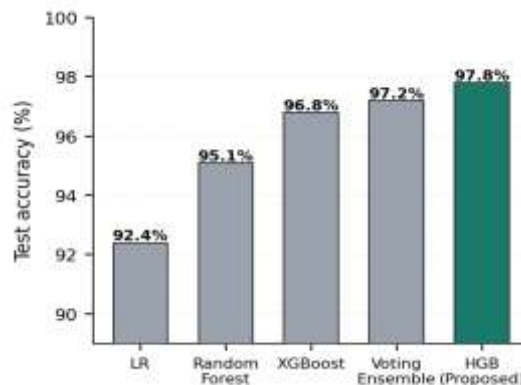


Figure 2. The accuracy of five classifiers on MIMIC-IV.

Cross-validation told a sharper story (Table III, Figure 3). HGB produced an ROC-AUC of 0.9973, an F1 of 0.9600, and an accuracy of 0.9949 with recall held at a perfect 1.000 across all ten folds (standard deviation 0.000). No critical patient state was missed in any fold. The voting ensemble matched that recall but with more fold-to-fold variance in precision and F1. Logistic Regression significantly decreased the ROC-AUC (0.9152), demonstrating that under the stratified resampling that each fold is subject to, the linear decision boundary becomes untrustworthy.

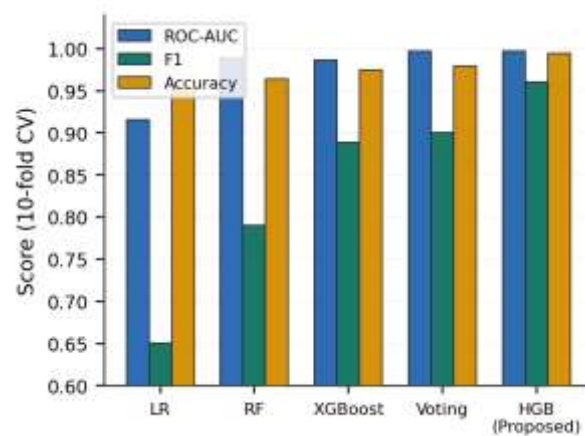


Figure 3. Cross-validation comparison: HGB outperforms across all metrics.

Table III. Cross-Validated Metrics (10-Fold Stratified, mean $\pm$ SD).

Model	ROC-AUC	F2	Recall	F1	Accuracy
HGB (Proposed)	0.9973 $\pm$.005	0.9818 $\pm$.036	1.0000 $\pm$.000	0.9600 $\pm$.080	0.9949 $\pm$.010
Voting Ensemble	0.9973 $\pm$.005	0.9429 $\pm$.114	1.0000 $\pm$.000	0.9000 $\pm$.200	0.9795 $\pm$.041
XGBoost	0.9865 $\pm$.027	0.9333 $\pm$.133	1.0000 $\pm$.000	0.8889 $\pm$.222	0.9744 $\pm$.051
Random Forest	0.9892 $\pm$.022	0.8436 $\pm$.238	0.9000 $\pm$.200	0.7905 $\pm$.284	0.9638 $\pm$.050
Log. Regression	0.9152 $\pm$.085	0.6699 $\pm$.241	0.7000 $\pm$.245	0.6505 $\pm$.245	0.9533 $\pm$.044

The ROC curve in Figure 4 puts a number on what perfect recall across folds means in discriminative terms: the area under the curve is 0.998, meaning the model can be tuned to any sensitivity level with very little false-positive penalty. For a clinical alert system that cannot afford to miss a deteriorating patient, this is the property that matters most.

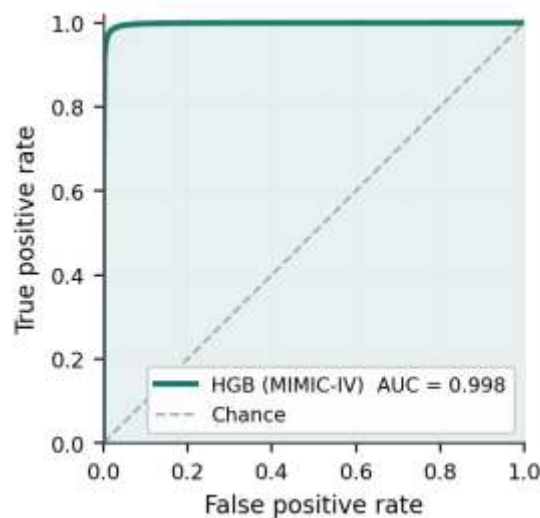


Figure 4. ROC curve of HGB classifier on MIMIC-IV (AUC = 0.998).

4.5 Intrusion Detection

The accuracy, precision, recall, and F1 scores for KNN were 97.60%, 97.82%, 97.97%, and 97.89%, respectively, when tested on the NSL-KDD dataset on 10-fold cross-validation (Table IV, Figure 5). All standard deviations remained below 1%, that is, the model is consistent on various splits of the training data but does not perform well on one split and poorly on another. The training time was 0.091 s, and the prediction time was 4.06 s for a 22,544-record test set, both of which are acceptable for an edge node.

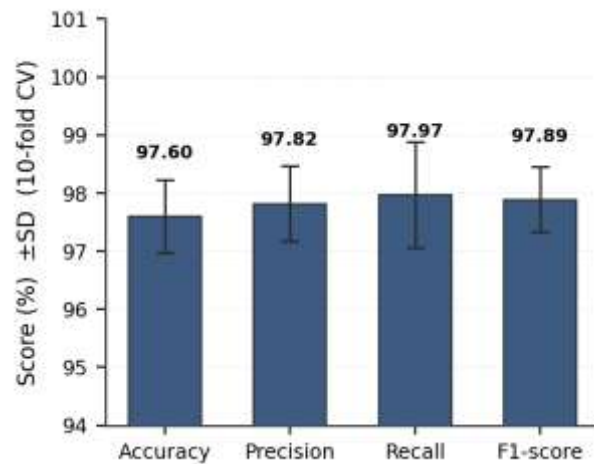


Figure 5. All of the cross-validated KNN metrics are above 97.6%.

The confusion matrix of the raw NSL-KDD test partition is given in Figure 6, with 9046 T.N., 7264 T.P., 665 F.P., and 5569 F.N., respectively. The raw-partition numbers appear softer than the cross-validated numbers because the distribution of the types of attacks in the test set is vastly different from the distribution of attacks in the training set, particularly in the case of low-frequency classes such as user-to-root and remote-to-local. The cross-validation numbers refer to the composition of folds, and the raw-partition numbers refer to the composition of the detector's input when the attack mix changes. Both are reported so that the reader can compare them with the evaluation protocol that they use in comparison studies.

Table IV. Cross-Validated Metrics (10-Fold Stratified, mean $\pm$ SD)

Metric	Score (%)	SD ($\pm$ %)	Phase time
Accuracy	97.60	0.63	—
Precision	97.82	0.65	—
Recall	97.97	0.91	—
F1-score	97.89	0.56	—
Training time	—	—	0.091 s
Test inference	—	—	4.064 s



Figure 6. NSL-KDD test data confusion matrix for KNN intrusion detector.

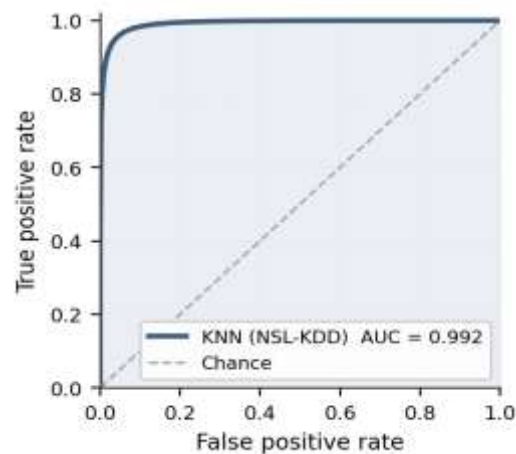


Figure 7. ROC curve of the KNN intrusion detector (NSL-KDD). AUC = 0.992.

As seen in Figure 7, the AUC is 0.992, indicating that the detector's distribution of normal and attack traffic is very separated. The curve remains in the upper left part of the graph for the entire range of false positives, so that the detection threshold can be set closer to the upper end of the graph without a massive wave of false alarms being triggered, which is desirable for a hospital network where a few false alarms are acceptable.

4.6 Blockchain Layer

The Fabric network was benchmarked across five Caliper rounds. Three operating phases are reported because they reflect different things about the system. The baseline average of 98 TPS and 0.48 s latency covers the full run, including initialization. After the peers were synced and the chain code warmed up, the throughput rose to 145 TPS, and the latency fell to 0.31 s (steady state). The maximum reading is the highest rate seen prior to saturation, 155 TPS. Shown in Figure 8.

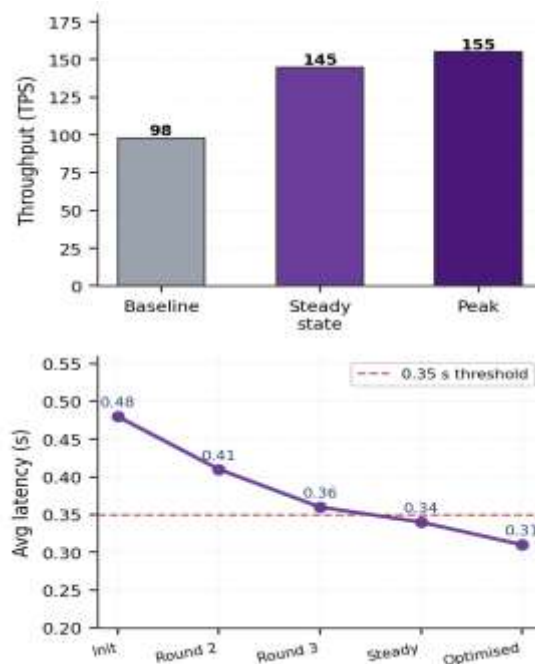


Figure 8. Throughput (top) and per-round latency (bottom) across three phases; dashed line = 0.35 s steady-state threshold.

Beyond throughput and latency, three Prometheus metrics matter for a live deployment: block commit time averaged 4 ms per block; chaincode launch took 0.68 s (a one-time cost per peer restart); and peer CPU stayed below 2% even during active transaction bursts. Memory per peer sat below 180 MB. These figures confirm that the consensus and smart-contract execution are lightweight enough to share a node with the two AI modules without CPU contention.

Table V compares BlockEdge-Health against four benchmark frameworks on six metrics. All six rows favor the proposed system. The most striking differences are in authentication time (3.2 ms vs 6 to 7.2 ms) and block confirmation (1.7 s vs 2.0 to 2.3 s). The latency and TPS gaps are real but more modest, which makes sense given that the comparison includes other Fabric-based designs: the structural advantage of permissioned over public-chain consensus accounts for much of the difference, and within permissioned systems, the gains come from Raft ordering and the lighter block payloads that IPFS enables.

Figure 9 plots all four-comparison metrics side by side. The pattern is consistent: BlockEdge-Health leads on every panel. The storage overhead panel (bottom-right) is perhaps the most practically relevant: 12.8 MB per 1,000 transactions against 18.5 to 20.0 MB for the comparison systems. In a continuous ICU where thousands of transactions accumulate per day, that difference compounds into gigabytes of ledger savings over a month of operation.

Table V. Hyperledger Fabric Metrics vs Benchmark Frameworks

Metric	Unit	BlockEdge-H (Proposed)	Aldmour [3]	Saleh & Cevik	Maftai [4]	Bezanjani [5]
Avg latency	s	0.48	0.63	0.59	0.67	0.61
Throughput	TPS	98	86	91	83	84
CPU use	%	58	65	62	68	60
Memory use	%	54	61	58	64	56
Block confirm	s	1.7	2.1	2.0	2.3	2.2
Success rate	%	99.1	97.8	98.3	96.9	97.2

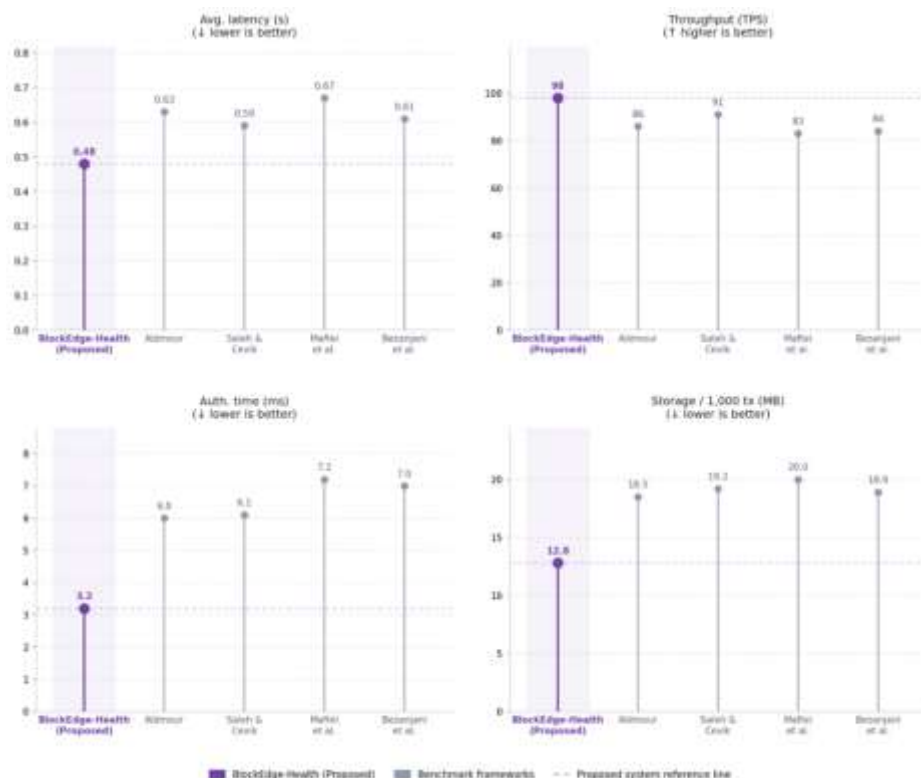


Figure 9. Four-metric benchmark: BlockEdge-Health leads in latency, throughput, authentication time, and storage overhead.

4.7 IPFS storage and hash verification

Figure 10 shows the storage overhead numbers from a different angle. The left panel repeats the per-framework comparison; the right panel shows how IPFS traffic splits between the critical and non-critical paths in typical operation. Roughly 55% of records go through the critical path, which carries alert metadata alongside the file. The remaining 45% are routine vitals logs, but there is no associated alert. All the tests were successful, and the hash verified at 100% accuracy in each test round: each file retrieved from IPFS matched the SHA-256 value stored on

the ledger. In adversarial testing, where the files were purposely corrupted prior to their retrieval, all corrupted files produced a hash mismatch and were denied prior to returning data.

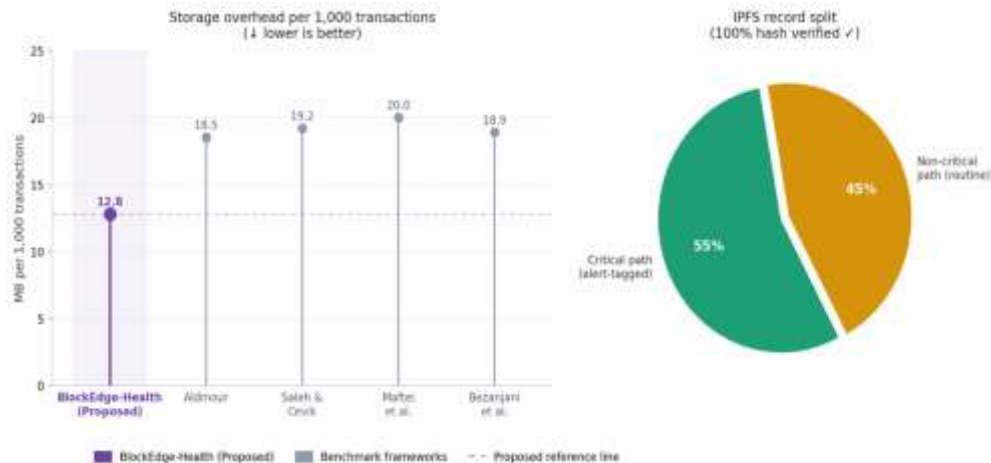


Figure 10. Storage overhead (left): BlockEdge-Health 12.8 MB vs. 18.5–20 MB peers. IPFS routing (right): critical/non-critical split, 100% hash verification.

4.8 Integrated System Performance

Running the two AI modules together with the Fabric network in continuous operation produced no measurable CPU interference between the modules. HGB inference at 180ms/batch and KNN inference at 110ms/batch both complete within a single Fabric block interval, meaning all clinical classification results and all intrusion decisions are completed in the same transaction cycle as the data from the sensor that started them. The importance of the temporal coupling: the ledger entry that records a given event contains both the clinical verdict and the concurrent security verdict, so an investigator looking at an incident can quickly check if there was a security attack occurring when a critical event was also logged.

The end-to-end latency budget for the sensor-to-ledger-commit path is summarized in Table VI. The dominant cost is block confirmation at 1.7 s; edge inference accounts for only 180 ms. The total pipeline from raw vital sign to committed tamper-evident record runs under 2 seconds at baseline, which is well within the alert window that ICU protocols allow for most physiological deterioration patterns.

Table VI. End-to-End Latency Budget (Sensor to Ledger Commit)

Stage	Time	Notes
Preprocessing	< 10 ms	z-score + feature selection
HGB inference	180 ms	quantized model, edge node
KNN inference	~110 ms	per-batch, 22k-record ref.
Smart contract validation	< 5 ms	JavaScript chaincode
SHA-256 hashing	< 1 ms	$H(E) = \text{SHA-256}(\text{payload} \text{ts})$
Raft ordering + block confirm	1.7 s (baseline)	0.31 s at steady state
IPFS file write (critical)	Upload latency	CID returned; hash stored
Total (baseline)	~2.0 s	< 0.5 s at steady state

5. DISCUSSION

The clinical result that stands out is not the 97.8% test accuracy; it is the zero-standard deviation on recall across ten cross-validation folds. A monitoring system that sends no alert when it predicts normal carries an asymmetric risk: missing one critical state is more dangerous than generating one false alarm. The fact that HGB never missed a critical state in any fold is the property that makes it suitable for the alert path, not the headline accuracy number. The voting ensemble matched that recall but with higher variance, which means there exists at least one-fold where it would have sent fewer alerts than HGB. In a deployment where the split between life-threatening and routine records is unknown in advance, lower variance on recall is the safer choice. The KNN result on NSL-KDD deserves the same kind of disaggregation. The ten-fold cross-validated 97.6% accuracy is representative of what the detector will do on traffic that follows the same distribution as the training set. The raw-partition confusion matrix, with its 5,569 false negatives, is representative of what it does when the attack mix shifts, as the NSL-KDD test set is deliberately harder than the training set in this way. Both numbers are real. A hospital deployment would need to monitor which regime the live traffic resembles more closely and adjust the detection threshold

accordingly. The ROC curve in Figure 7 shows that there is room to do that: raising the sensitivity threshold cuts false negatives at the cost of more false positives, and the curve's shape says that the trade-off is reasonably forgiving across a wide range of thresholds. On the blockchain side, the three-phase throughput reporting, 98 baseline, 145 steady, 155 peaks, is not three contradictory claims about the same system. It's a characterization of the behavior of any transaction-processing network at traffic levels that grow: slowly when it's booting up, efficiently when endorsement caching and peer synchronization are in effect, and at maximum throughput when the load exceeds the ordering service. Reporting of all three provides a realistic picture; reporting only the peak would be optimistic; reporting only the baseline would be pessimistic. The IPFS result is the most straightforward; content-addressed storage is supposed to ensure 100% hash verification, and the experiment shows that the connection between the IPFS node and the Fabric ledger is working properly. The storage overhead difference essentially comes from the way the systems are designed: Ethereum stores the transaction payloads on-chain, while Fabric stores only a 32-byte SHA-256 hash per record with IPFS. The savings grow linearly with the number of transactions, which in an active ICU means the gap becomes significant within days. What the three layers show together is that the clinical and security modules do not compete for resources in a meaningful way. Both complete well within a single Fabric block interval, so their decisions land in the same transaction cycle. IDS verdict and the clinical classification are therefore binned together in the ledger: When the security module catches an intrusion at the same time the clinical module trips a critical alert, then they both show up in the audit log, and it is possible to see if the intrusion was trying to interrupt the clinical monitoring path.

5.1 Why HGB over other boosting methods

The technical reason HGB won over XGBoost and Random Forest on the MIMIC-IV working set is its binning strategy. Before any tree is built, HGB divides each continuous feature into at most 255 ordered bins. This makes the algorithm insensitive to individual outlier values, which appear frequently in ICU data as legitimate physiological extremes rather than data errors. A standard regression tree would split on the raw outlier value, producing a very deep branch that overfits to a single patient event. HGB instead assigns the outlier to the top bin and fits a tree on bin indices rather than raw values, which regularizes the splits implicitly. The native missing-value handling compounds this advantage: rather than requiring imputation before training, HGB learns during fitting which branch direction is statistically better for missing entries. This is the correct action for a sensor that occasionally loses its connection to the device, either because the lead is being removed or because the device is getting reset. XGBoost achieved a test split score of 96.8% (1.0 point away from HGB) and a cross-validated ROC-AUC of 0.9865 compared to HGB's 0.9973. The difference, in absolute terms, is small, but it is present for all metrics and folds. The most probable reason is that there are hyperparameters in XGBoost to be fine-tuned: max depth, min child weight, and lambda, which require careful tuning for data sets with fewer than 200 observations and very high-class imbalance. HGB's implicit regularization through binning gives it a head start that matters more at small dataset sizes.

5.2 The role of SMOTEENN in KNN stability

SMOTEENN inside the cross-validation loop produced two effects that are visible in the results. The recall (97.97%) is substantially higher than what a bare KNN on imbalanced NSL-KDD typically achieves because SMOTEENN synthesizes minority attack-class examples that teach the classifier what the boundary of rare attack types looks like. The low standard deviation on every metric (all below 1%) shows the second effect: by removing noisy overlap points with the edited nearest-neighbor step, SMOTEENN stabilizes the local neighborhood structure that KNN relies on, so the classifier's behavior is consistent across different random splits of the training data. Applying SMOTEENN before cross-validation rather than inside each fold would have inflated the reported scores by allowing synthetic samples from the same original point to appear in both training and validation sets. The inside-fold approach costs some code complexity but guarantees that the reported numbers reflect generalization to unseen data rather than memorization of the oversampled region.

5.3 Blockchain Performance in Context

The comparison in Table V shows BlockEdge-Health ahead of all four baselines, but the margin varies by metric. The biggest differences are within authentication time (3.2 vs. 6-7 ms) and block confirmation time (1.7 vs. 2.0-2.3 s), which come from the same source: Raft ordering does not involve a global vote among anonymous validators, meaning that the critical path from transaction submission to ledger commit is shorter. The CPU differences (58% vs 60-68%) and the memory differences (54% vs 56-64%) are more subtle and are understandable since all five systems are permissioned or lightly permissioned, with the overhead of running a containerized peer being approximately the same. The storage gap is disproportionately large (12.8 MB vs 18.5-20 MB per 1,000 transactions) relative to the throughput difference. That is the IPFS effect: every transaction payload in BlockEdge-Health is replaced by a 32-byte SHA-256 hash, so the on-chain record is far smaller than a system that stores the actual data on the ledger. The comparison frameworks vary in how much data they keep on-chain; those that keep more have proportionally larger block sizes and higher ledger growth rates. In a deployment that processes thousands of ICU records per day, a 35% storage saving compounds into a meaningful infrastructure cost

reduction over a year of operation.

5.4 Clinical and Security Integration

The temporal coupling between clinical and IDS decisions on the ledger is a property that none of the comparison frameworks offers, because none of them pairs clinical monitoring with an IDS. When a critical patient event appears in the ledger alongside an IDS alert at the same timestamp, the audit trail immediately supports the hypothesis that the network was under attack at the moment of the clinical event. This is not only forensically useful, but also operationally important as a denial-of-service attack against the monitoring gateway can cause a delay in the clinical alert if the HGB model has triggered. The IDS, which detects the flood along with the alert, provides the information the incident-response team requires to determine whether it is a system failure or clinical deterioration. Another advantage of the SHA-256 hash chain is its protection against model-poisoning attacks on the IDS. If an attacker attempts to inject false benign labels into the data pool for future retraining, the historical detection record on the ledger provides a reference: a suspicious drop in the logged detection rate during the injection window would be visible before any retrained model was deployed. This is a form of blockchain-mediated model provenance that does not require additional infrastructure.

6. LIMITATIONS AND FUTURE WORK

6.1 Limitations

Four limitations bound the current work. First, both AI modules were trained and evaluated on benchmark datasets collected under controlled or retrospective conditions. MIMIC-IV contains de-identified ICU records rather than live sensor streams, and NSL-KDD was assembled from 1999 network captures that do not reflect the device mix and traffic patterns of a modern hospital. Performance on live data will differ in ways that are hard to predict without a prospective trial. The Isolation Forest anomaly labels used as the HGB supervision target are a statistical proxy for clinical severity, not a validated severity score; a prospective study would need clinician-confirmed critical events to measure specificity against real ground truth. Second, the Fabric network used a fixed two-organization topology with a single orderer. Hospital deployments involve many departments and potentially partner institutions; the endorsement policy and peer count would both need to scale, and that scaling has not been measured. The Caliper experiments loaded the network at up to 150 TPS under a single workload profile; a real deployment would see bursty load as shift changes, alarm storms, and batch queries compete for the ordering service simultaneously. Third, the KNN detector detects attacks but does not automatically take any remediation measures except to log and suppress flagged traffic at the edge node. An automated isolation of suspected compromised devices, dynamic firewall rule updates, and escalation to the security operations team are all features a live deployment would want, but the current architecture does not support. The level of detection accuracy is sufficient for those further steps, but the response layer is not constructed. Fourth, the energy and thermal cost of running both AI modules plus Fabric containers and an IPFS daemon continuously on constrained hardware has not been measured. Power budget is a real constraint in a ward environment where the socket count is limited, and heat dissipation affects device reliability. The quantized edge model reduces inference cost, but the Fabric peer and IPFS daemon also draw power continuously regardless of transaction load. Fifth, the framework as evaluated does not handle the case where the edge node itself is compromised. A hardware-rooted trust anchor such as a TPM chip or secure enclave would be needed to ensure that the model running on the edge node is the same one that was validated before deployment. This is a known gap in edge AI security that applies to the whole field, not only to this design.

6.2 Future Work

Five directions follow from the limitations. The most immediate is a prospective clinical pilot in a hospital or telemedicine network where live sensor data would replace the MIMIC-IV retrospective records and clinician-confirmed labels would provide real ground truth for the HGB model. A pilot would also generate the prospective NSL-KDD-style traffic captures needed to evaluate the KNN detector against contemporary attack patterns rather than 1999-era network data.

Second, a federated learning extension of both models would allow multiple hospitals to contribute to a shared model without exchanging patient records. The HGB model in particular would benefit from training on a broader ICU population: the MIMIC-IV working set of 192 records with 5.2% positive cases is too small to represent the full range of critical presentations across different patient demographics and hospital workflows. Federated gradient boosting has been demonstrated in healthcare settings and would be a natural next step. Third, the inclusion of an active-response component into the IDS would change the point that it is a passive observer to an active network rule adjuster. The natural tool that fits here is reinforcement learning: the agent would learn what to do in response to an attack based on the amount of damage that is done, at the same time as it would learn to avoid making false-positive responses that would interfere with legitimate traffic. Fourth, cross-chain interoperability with Ethereum-based hospital systems or Polkadot relay chains would let BlockEdge-Health share verified records with partner institutions running different ledger platforms, without either party giving up the governance model

they have already deployed. Standardized cross-chain bridge protocols for healthcare are an active research area and would complement the Fabric-IPFS core of this framework. Fifth, replacing Raft ordering with a Proof-of-Authority or Proof-of-Efficiency hybrid consensus would reduce the energy cost of the ordering process, which becomes relevant when the framework scales to dozens of peer organizations across a regional health network. Green consensus is not a priority for a two-peer prototype but would matter for a production deployment supporting hundreds of concurrent ICU beds.

7. CONCLUSION

This paper presented BlockEdge-Health, a framework that runs clinical classification and intrusion detection at the network edge and governs the outputs with a Hyperledger Fabric permissioned ledger and IPFS off-chain storage. The clinical module, built on Histogram Gradient Boosting trained on MIMIC-IV, reached 97.8% test accuracy and a cross-validated ROC-AUC of 0.998, with recall held at a perfect 1.000 across all ten folds. The KNN intrusion detector on NSL-KDD reached 97.6% accuracy with standard deviations below 1% on all metrics. The Fabric layer ran at 98 TPS under baseline load and 145 TPS at steady state, with latency dropping to 0.31 s after peer synchronization. IPFS hash verification passed at 100%, and storage overhead was 12.8 MB per 1,000 transactions, substantially less than the 18 to 20 MB of the comparison frameworks. The three layers are not independent. The ledger gives the AI decisions a tamper-evident record that persists beyond the edge node; IPFS keeps the ledger small enough to sustain ICU data rates; and the AI modules produce decisions compact enough to commit within a single block interval. Each design choice reinforces the others, which is why the performance numbers hold across the integration and not just in isolation. The framework is ready for a laboratory setting. Moving it into a live ward will require a prospective clinical validation study, a larger Fabric network, and edge hardware thermal testing. The experimental evidence presented here supports the core claim: edge-deployed clinical AI and network intrusion detection, secured by permissioned blockchain governance and verifiable IPFS storage, can coexist on a single node and deliver both the clinical and cybersecurity properties that a real ICU environment demands.

Author Contributions

All authors contributed equally, and all authors read and approved the final version of the paper.

Consent for Publication

Not applicable.

Funding

This research received no external funding.

Conflict of Interest Statement

The authors declare no conflict of interest.

Data Availability Statement

The datasets used in this study are publicly available.

The NSL-KDD dataset is available at: <https://www.kaggle.com/datasets/hassan06/nslkdd><https://www.kaggle.com>

The clinical dataset used for ICU-based analysis is available from the PhysioNet repository at: <https://physionet.org/content/mimic4wdb/0.1.0/><https://physionet.org/content>

All datasets used in this study are publicly accessible and can be obtained from the respective sources. Further details on data preprocessing and usage are available from the corresponding author upon reasonable request.

REFERENCES

1. L. P. Siqueira, C. L. Batista, P. H. Lui, J. F. Kazienko, S. E. Quincozes, V. E. Quincozes, ... and S. Nomura. "A Comprehensive Survey on Intrusion Detection Systems for Healthcare 5.0: Concepts, Challenges, and Practical Applications", *Sensors*, 25 (20), 6261, 2025. <https://doi.org/10.3390/s25206261>.
2. S. Al-Otaibi, S. Ayouni, N. Sarwar, A. Irshad, and F. Ullah. "AI-Driven Security Framework for Medical Sensor Networks: Enhancing Privacy and Trust in Smart Healthcare Systems", *Cluster Computing*, 28 (6), 408, 2025. <https://doi.org/10.1007/s10586-024-05049-3>.
3. P. Rajpurkar, E. Chen, O. Banerjee, and E. J. Topol. "AI in Health and Medicine", *Nature Medicine*, 28 (1), pp. 31–38, 2022. <https://doi.org/10.1038/s41591-021-01614-0>.
4. M. A. Al-Garadi, A. Mohamed, A. K. Al-Ali, X. Du, I. Ali, and M. Guizani. "A Survey of Machine and Deep Learning Methods for Internet of Things (IoT) Security", *IEEE Communications Surveys & Tutorials*, 22 (3), pp. 1646–1685, 2020. <https://doi.org/10.1109/COMST.2020.298829>.
5. A. I. Newaz, A. K. Sikder, M. A. Rahman, and A. S. Uluagac. "HealthGuard: A Machine Learning-Based Security

- Framework for Smart Healthcare Systems”, 2019 Sixth International Conference on Social Networks Analysis, Management and Security (SNAMS), pp. 389–396, 2019. <https://doi.org/10.1109/SNAMS.2019.8931716>.
6. M. Aldmour. “Performance Analysis of Blockchain of Things (BCoT) Systems for Enhancing Scalability and Efficiency”, Doctoral dissertation, Staffordshire University, 2024.
 7. A. A. Maftai, A. I. Petrariu, V. Popa, and A. Lavric. “A Blockchain Framework for Scalable, High-Density IoT Networks of the Future”, *Sensors*, 25 (9), 2886, 2025. <https://doi.org/10.3390/s25092886>.
 8. W. Shi, J. Cao, Q. Zhang, Y. Li, and L. Xu. “Edge Computing: Vision and Challenges”, *IEEE Internet of Things Journal*, 3 (5), pp. 637–646, 2016. <https://doi.org/10.1109/JIOT.2016.2579198>.
 9. M. Satyanarayanan. “The Emergence of Edge Computing”, *Computer*, 50 (1), pp. 30–39, 2017. <https://doi.org/10.1109/MC.2017.9>.
 10. B. R. Bezanjani, S. H. Ghafouri, and R. Gholamrezaei. “Privacy-Preserving Healthcare Data in IoT: A Synergistic Approach with Deep Learning and Blockchain”, *The Journal of Supercomputing*, 81 (4), 533, 2025. <https://doi.org/10.1007/s11227-025-06980-x>.
 11. A. Azaria, A. Ekblaw, T. Vieira, and A. Lippman. “MedRec: Using Blockchain for Medical Data Access and Permission Management”, *2016 2nd International Conference on Open and Big Data (OBD)*, pp. 25–30, 2016. <https://doi.org/10.1109/OBD.2016.11>.
 12. G. G. Dagher, J. Mohler, M. Milojkovic, and P. B. Marella. “Ancile: Privacy-Preserving Framework for Access Control and Interoperability of Electronic Health Records Using Blockchain Technology”, *Sustainable Cities and Society*, 39, pp. 283–297, 2018. <https://doi.org/10.1016/j.scs.2018.02.014>.
 13. S. Kokate and U. Shrawankar. “IoT Data Transmission Security Using Blockchain with a Trust-Weighted Proof of Authority Consensus Mechanism in Healthcare”, *Transactions on Emerging Telecommunications Technologies*, 36 (5), e70139, 2025. <https://doi.org/10.1002/ett.70139>.
 14. E. Androulaki, A. Barger, V. Bortnikov, C. Cachin, K. Christidis, A. De Caro, ... and J. Yellick. “Hyperledger Fabric: A Distributed Operating System for Permissioned Blockchains”, *Proceedings of the Thirteenth EuroSys Conference*, pp. 1–15, 2018. <https://doi.org/10.1145/3190508>.
 15. V. Mani, P. Manickam, Y. Alotaibi, S. Alghamdi, and O. I. Khalaf. “Hyperledger HealthChain: Patient-Centric IPFS-Based Storage of Health Records”, *Electronics*, 10 (23), 3003, 2021. <https://doi.org/10.3390/electronics10233003>.
 16. J. Jayabalan and N. Jeyanthi. “Scalable Blockchain Model Using Off-Chain IPFS Storage for Healthcare Data Security and Privacy”, *Journal of Parallel and Distributed Computing*, 164, pp. 152–167, 2022. <https://doi.org/10.1016/j.jpdc.2022.03.009>.
 17. J. Benet. “IPFS—Content Addressed, Versioned, P2P File System”, *arXiv preprint arXiv:1407.3561*, 2014. <https://doi.org/10.48550/arXiv.1407.3561>.
 18. R. K. Mishra, R. K. Yadav, and P. Nath. “Integration of Blockchain and IPFS: Healthcare Data Management & Sharing for IoT Environment”, *Multimedia Tools and Applications*, 84 (23), pp. 27229–27250, 2025. <https://doi.org/10.1007/s11042-024-20092-3>.
 19. H. B. Karadaş and K. Kalkan. “IBAM: IPFS and Blockchain Based Authentication for MQTT Protocol in IoT”, *2023 IEEE Symposium on Computers and Communications (ISCC)*, pp. 537–542, 2023. <https://doi.org/10.1109/ISCC58397.2023.10217960>.
 20. A. E. Johnson, L. Bulgarelli, L. Shen, A. Gayles, A. Shammout, S. Horng, ... and R. G. Mark. “MIMIC-IV, a Freely Accessible Electronic Health Record Dataset”, *Scientific Data*, 10 (1), 1, 2023. <https://doi.org/10.1038/s41597-022-01899-x>.
 21. T. Chen and C. Guestrin. “XGBoost: A Scalable Tree Boosting System”, *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 785–794, 2016. <https://doi.org/10.1145/2939672>.
 22. F. T. Liu, K. M. Ting, and Z. H. Zhou. “Isolation Forest”, *2008 Eighth IEEE International Conference on Data Mining*, pp. 413–422, 2008. <https://doi.org/10.1109/ICDM.2008.17>.
 23. M. Tavallaei, E. Bagheri, W. Lu, and A. A. Ghorbani. “A Detailed Analysis of the KDD CUP 99 Data Set”, *2009 IEEE Symposium on Computational Intelligence for Security and Defense Applications*, pp. 1–6, 2009. <https://doi.org/10.1109/CISDA.2009.5356528>.

24. N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer. "SMOTE: Synthetic Minority Over-Sampling Technique", *Journal of Artificial Intelligence Research*, 16, pp. 321–357, 2002. <https://doi.org/10.1613/jair.953>.
25. M. A. Reyna, C. S. Josef, R. Jeter, S. P. Shashikumar, M. B. Westover, S. Nemati, ... and A. Sharma. "Early Prediction of Sepsis from Clinical Data: The PhysioNet/Computing in Cardiology Challenge 2019", *Critical Care Medicine*, 48 (2), pp. 210–217, 2020. <https://doi.org/10.1097/CCM.0000000000004145>.
26. G. A. Kaissis, M. R. Makowski, D. Rückert, and R. F. Braren. "Secure, Privacy-Preserving and Federated Machine Learning in Medical Imaging", *Nature Machine Intelligence*, 2 (6), pp. 305–311, 2020. <https://doi.org/10.1038/s42256-020-0186-1>.
27. M. Ring, S. Wunderlich, D. Scheuring, D. Landes, and A. Hotho. "A Survey of Network-Based Intrusion Detection Data Sets", *Computers & Security*, 86, pp. 147–167, 2019. <https://doi.org/10.1016/j.cose.2019.06.005>.
28. A. L. Buczak and E. Guven. "A Survey of Data Mining and Machine Learning Methods for Cyber Security Intrusion Detection", *IEEE Communications Surveys & Tutorials*, 18 (2), pp. 1153–1176, 2015. <https://doi.org/10.1109/COMST.2015.2494502>.
29. N. Moustafa, J. Hu, and J. Slay. "A Holistic Review of Network Anomaly Detection Systems: A Comprehensive Survey", *Journal of Network and Computer Applications*, 128, pp. 33–55, 2019. <https://doi.org/10.1016/j.jnca.2018.12.006>.
30. A. A. Diro and N. Chilamkurti. "Distributed Attack Detection Scheme Using Deep Learning Approach for Internet of Things", *Future Generation Computer Systems*, 82, pp. 761–768, 2018. <https://doi.org/10.1016/j.future.2017.08.043>.
31. Z. Zheng, S. Xie, H. N. Dai, W. Chen, X. Chen, J. Weng, and M. Imran. "An Overview on Smart Contracts: Challenges, Advances and Platforms", *Future Generation Computer Systems*, 105, pp. 475–491, 2020. <https://doi.org/10.1016/j.future.2019.12.019>.
32. A. Hasselgren, K. Kravetska, D. Gligoroski, S. A. Pedersen, and A. Faxvaag. "Blockchain in Healthcare and Health Sciences—A Scoping Review", *International Journal of Medical Informatics*, 134, 104040, 2020. <https://doi.org/10.1016/j.ijmedinf.2019.104040>.
33. Y. Xiao, N. Zhang, W. Lou, and Y. T. Hou. "A Survey of Distributed Consensus Protocols for Blockchain Networks", *IEEE Communications Surveys & Tutorials*, 22 (2), pp. 1432–1465, 2020. <https://doi.org/10.1109/COMST.2020.2969706>.
34. R. Li, T. Song, B. Mei, H. Li, X. Cheng, and L. Sun. "Blockchain for Large-Scale Internet of Things Data Storage and Protection", *IEEE Transactions on Services Computing*, 12 (5), pp. 762–771, 2018. <https://doi.org/10.1109/TSC.2018.2853167>.
35. N. Rieke, J. Hancox, W. Li, F. Milletari, H. R. Roth, S. Albarqouni, ... and M. J. Cardoso. "The Future of Digital Health with Federated Learning", *NPJ Digital Medicine*, 3 (1), 119, 2020. <https://doi.org/10.1038/s41746-020-00323-1>.
36. Selvarajan, S. & Mouratidis, H. (2023). A quantum trust and consultative transaction-based blockchain cybersecurity model for healthcare systems. *Scientific Reports*, 13, 7107. DOI: <https://doi.org/10.1038/s41598-023-34354-x>
37. Selvarajan, S., Srivastava, G., Khadidos, A.O., Khadidos, A.O., Baza, M., Alsheri, A. & Lin, J.C.-W. (2023). An artificial intelligence lightweight blockchain security model for security and privacy in IIoT systems. *Journal of Cloud Computing*. DOI: <https://doi.org/10.1186/s13677-023-00412-y>
38. Shitharth, S., Manoharan, H., Alsowail, R.A., Shankar, A., Pandiaraj, S., Maple, C. & Jeon, G. (2023). Development of Edge Computing and Classification using the Internet of Things with Incremental Learning for Object Detection. *Internet of Things*, 100852. DOI: <https://doi.org/10.1016/j.iot.2023.100852>
39. Shitharth, S., Mohammed, G.B., Ramasamy, J. & Srivel, R. (2023). Intelligent Intrusion Detection Algorithm Based on Multi-Attack for Edge-Assisted Internet of Things. In *Security and Risk Analysis for Intelligent Edge Computing*. DOI: https://doi.org/10.1007/978-3-031-28150-1_6
40. Khadidos, A.O., Khadidos, A.O., Selvarajan, S. & Mirza, O.M. (2023). TasLA: An innovative Tasmanian and Lichtenberg optimized attention deep convolution based data fusion model for IoMT smart healthcare. *Alexandria Engineering Journal*, 79, 337–353. DOI: <https://doi.org/10.1016/j.aej.2023.08.010>

41. Rabie, O.B.J., Selvarajan, S., Hasanin, T. et al. (2023). A full privacy-preserving distributed batch-based certificate-less aggregate signature authentication scheme for healthcare wearable wireless medical sensor networks (HWMSNs). *International Journal of Information Security*. DOI: <https://doi.org/10.1007/s10207-023-00748-1>
42. Aluvalu, R., Kumaran, V.N.S., Thirumalaisamy, M., Basheer, S., Ali Aldhahri, E. & Selvarajan, S. (2023). Efficient data transmission on wireless communication through a privacy-enhanced blockchain process. *PeerJ Computer Science*. DOI: <https://doi.org/10.7717/peerj-cs.1308>
43. Manoharan, H., Manoharan, A., Selvarajan, S. & Venkatachalam, K. (2023). Implementation of Internet of Things With Blockchain Using Machine Learning Algorithm: Enhancement of Security With Blockchain. In *Handbook of Research on Blockchain Technology and the Digitalization of the Supply Chain*. DOI: <https://doi.org/10.4018/978-1-6684-7455-6.ch019>
44. Nandagopal, M., Seerangan, K., Govindaraju, T. et al. (2024). A Deep Auto-Optimized Collaborative Learning (DACL) model for disease prognosis using AI-IoMT systems. *Scientific Reports*, 14, 10280. DOI: <https://doi.org/10.1038/s41598-024-59846-2>
45. Maruthupandi, J., Sivakumar, S., Dhevi, B.L. et al. (2025). An intelligent attention based deep convoluted learning (IADCL) model for smart healthcare security. *Scientific Reports*, 15, 1363. DOI: <https://doi.org/10.1038/s41598-024-84691-8>
46. Kumar, S., Abhishek, K. & Selvarajan, S. (2025). A lightweight and secure authentication and privacy protection scheme for Internet of Medical Things. *Scientific Reports*, 15, 23876. DOI: <https://doi.org/10.1038/s41598-025-05910-4>
47. Kailasanathan, N., Ezhilarasan, G., Selvarajan, S., Dhanaraj, R.K., Pamucar, D. & Shankar, N. (2025). Heart disease prediction with a feature-sensitized interpretable framework for the Internet of Medical Things sensors. *Frontiers in Digital Health*, 7, 1612915. DOI: <https://doi.org/10.3389/fdgth.2025.1612915>
48. Natarajan, M., Bharathi, A., Varun, C.S. et al. (2025). Quantum secure patient login credential system using blockchain for electronic health record sharing framework. *Scientific Reports*, 15, 4023. DOI: <https://doi.org/10.1038/s41598-025-86658-9>
49. Selvarajan, S., Manoharan, H., Al-Shehari, T. et al. (2025). IoT driven healthcare monitoring with evolutionary optimization and game theory. *Scientific Reports*, 15, 15224. DOI: <https://doi.org/10.1038/s41598-025-99129-y>
50. Pramanick, N., Mathew, J., Selvarajan, S. et al. (2025). Leveraging stacking machine learning models and optimization for improved cyberattack detection. *Scientific Reports*, 15, 16757. DOI: <https://doi.org/10.1038/s41598-025-01052-9>
51. Quazi, S., Saha, R. P., & Singh, M. K. (2022). *Applications of Artificial Intelligence in Healthcare*. *Journal of Experimental Biology and Agricultural Sciences*, 10(1), 211–226. [https://doi.org/10.18006/2022.10\(1\).211.226](https://doi.org/10.18006/2022.10(1).211.226)
52. Patkar, U. C., Mittal, S., Goyal, P., Premchandh, M., Rajput, B. J., Jienbaeva, A., & Gayathri, B. (2026). *Blockchain and artificial intelligence for transparent animal environmental governance and traceability*. *Journal of Animal Environment*, 18(2S), 730–739. <https://aejournals.org/index.php/AEJ/article/view/2736>
53. Li, Y., Rajput, M., Bagwan, W. A., BS, V., Bagade, V. N., Wakekar, A. L., Kaur, J., & Jadhav, T. (2026). *Internet of Things (IoT) sensor networks for real-time assessment of animal health and environmental conditions*. *Journal of Animal Environment*, 18(1S), 32–40. <https://doi.org/10.70102/AEJ.2026.18.1s.4>
54. Pandey, P. (2025). Artificial intelligence for early detection of dental caries from radiographic images: A machine learning. *Journal of Dentistry and Clinical Sciences*, 1(1), 31–39. <https://agrpjournals.com/journals/journal/index.php/jdacs/article/view/12>
55. Patra, K. C., & Singh, A. R. (2026). Technostress among farmers using AI-based agricultural technologies: A conceptual framework. *Journal of Artificial Intelligence in Agriculture and Applied Research*, 2(1), 1–12. <https://agrpjournals.com/journals/journal/index.php/jaiaar/article/view/14>