BULLETIN OF THE POLISH ACADEMY OF SCIENCES
TECHNICAL SCIENCES, Vol. 74(6), 2026, Article number: 1956
DOI: 10.24425/bpasts.2026.1956

Enhancing security in edge computing healthcare analysis by integration of deep learning

Kottapalli Vijaya Naga Valli* **and Sujihelen L**

Sathyabama Institute of Science and Technology, India

Abstract. Healthcare monitoring (HM) enables individuals to track their personal health with the help of different smart devices. It employs IoT technologies and cloud architecture for data collection, transmission via smart devices, storage, analysis and healthcare recommendations. However, it is quite difficult to process so much data from so many IoT devices in short periods of time. Thus, technology frameworks, such as edge computing, might serve as a layer of intermediary communication between users and the cloud. These systems still have drawbacks, though, including poor security, excessive latency, wasteful energy use, inefficient bandwidth use and a lack of automation. In this study, a secure healthcare monitoring system (Secure-HMS) in edge computing environment using deep learning networks has been proposed. The suggested system employs a combination of fully homomorphic quantum Diffie-Hellman encryption to protect patient health data during transmission and to manage it securely. The suggested method shares the key to convert the plain data into cipher data using the innovative secure quantum Diffie-Hellman key exchange (QDHKE) technique, then uses FH operations on the encrypted data before sending it to the edge server. Furthermore, multi-head attention Bi-GRU (MA-BGRU) is utilized for prediction and enhancing overall system efficiency. This system mainly serves for real-time health monitoring, controlling the security, predictability and confidentiality of medical data. The experimental outcomes demonstrate that the Secure-HMS framework's overall security rate is 4.08%, 3.06% and 2.04% better than those of MEdge-chain, EBPPA and SDN-based edge networks, respectively.

Keywords: edge computing; healthcare; deep learning; encryption; fully homomorphic.

1. INTRODUCTION

Every individual deserves timely access to healthcare as a vital part of their well-being. A country's healthcare system is a key indicator of its development and advancement. Yet, it is extremely difficult to deliver high-quality, consistent healthcare in the case of the nation's large population or wide range of living situations [1]. Traditional healthcare methods often fall short when attempting to provide real-time monitoring, leading to delays in diagnosis and treatment [2]. This has driven the adoption of smart healthcare systems, which utilize advanced sensors and IoT devices to enable continuous health tracking, offering personalized care and timely intervention [3]. It provides healthcare to persons who are unable to visit the clinic by continually monitoring vital signs with wearable IoT (Internet-of-Things) sensors, which is especially useful for chronic and geriatric patients [4]. A report published by Statista in August 2022 claims that the use of smart healthcare is growing across several healthcare industries [5].

However, smart healthcare systems face significant challenges, including data privacy, latency and processing limitations, which can all affect the reliability and accessibility of healthcare services [6]. The transmission of biological data gathered by IoT devices to third-party cloud organizations poses difficulties for patient privacy and data security protection in the healthcare industry [7]. The need for secure data handling is crucial, as sensitive patient information is vulnerable to breaches and unauthorized access [8]. Integrating edge computing in healthcare enhances system responsiveness and strengthens the protection of patient data, making it essential for modern, secure healthcare infrastructures [9, 10].

While edge computing presents significant advantages for secure healthcare monitoring, it also faces key limitations. A major concern is the lack of robust security measures, which can expose sensitive patient data to potential breaches [11, 12]. Additionally, high latency issues may arise during data processing and transmission, affecting the timeliness of critical healthcare interventions. Energy inefficiency in edge devices can lead to increased operational costs, while inefficient bandwidth utilization hampers the effectiveness of data transfer [13, 14]. Lastly, a shortage of automaticity in managing and processing data can limit the system's ability to respond swiftly to changing healthcare needs, potentially impacting overall patient care [15]. To overcome these issues, in this study, a secure healthcare monitoring system (Secure-HMS) in an edge computing environment using deep learning networks has been introduced.

The key objectives of the Secure-HMS model are as follows:

- Initially, the data acquisition layer collects the healthcare data from sensors in different healthcare devices. The data are subsequently transmitted to the edge layer via the gateway devices.
- In the edge computing layer, medical data encryption is performed via fully homomorphic quantum Diffie-Hellman encryption, ensuring patient data confidentiality and integrity.

*e-mail: kvnv@srkrec.ac.in

Manuscript submitted 2025-03-31, revised 2025-07-21, initially accepted for publication 2026-07-24, published in November 2026.

© 2026 The Author(s). This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/)

- In the data accessing layer, the authentication and the encrypted data will be decrypted and prediction will be performed using multi-head attention-based Bi-GRU and sent to the healthcare community.
- The efficiency of the Secure-HMS model is analyzed based on encryption and decryption time, security analysis, delay, packet delivery ratio (PDR), throughput, average response time and average control overhead.

The remaining sections of this work are arranged as follows: Section 2 details the related work while Section 3 explains the Secure-HMS methodology. Section 4 presents the experimental outcomes and, finally, Section 5 details the conclusion and future work.

2. LITERATURE SURVEY

Medical healthcare data sharing has been considered a crucial issue during data outsourcing because the privacy of medical data is necessary. Therefore, many researchers have concentrated on secure data sharing and provided many solutions for data security. This section discusses several works.

In 2020, J. Li *et al.* [16] developed an SDN-integrated edge computing for IoT-enabled healthcare systems. IoT devices collect patient data and transmit them to edge servers for processing, storing and analysis after edge servers authenticate them using a lightweight approach. The suggested model attains enhanced security and efficient data handling for healthcare systems. However, the trust in SDN may introduce potential scalability and complexity issues.

In 2021 M.S. Christo *et al.* [17] suggested an elliptic curve cryptography (ECC) to share the data efficiently. The suggested system ensures data authentication through 512-bit encryption, with blockchain providing confidentiality by granting access to ethical users. Encrypted data are stored on edge devices for fast access, while authenticated users can decrypt, process and store updated data in both the blockchain and cloud server. The suggested model improves medical data security through edge devices; however, it may encounter challenges related to computational overhead and resource limitations.

In 2021 A.A. Abdellatif *et al.* [18] presented a MEdge-chain framework based on blockchain for secure edge computing in healthcare. The suggested design employs blockchain technology to maintain user data transfer records and handle the number of blockchain transactions to guarantee data security. By allowing consumers to keep an eye on their data transactions, user-centricity is accomplished. The suggested model enhances the efficiency of medical data exchange. However, it may encounter issues related to interoperability.

In 2021, A. Singh and K. Chatterjee [19] developed a smart healthcare technology based on an edge computing network. Using privacy-preserving searchable encryption (PPSE), the edge computing layer in this architecture handles network latency and protects patient data privacy. It also implements access control to prevent unauthorized access to data stored remotely. The suggested approach attains improved data security and processing efficiency while it may face limitations in managing large-scale data.

In 2022, R. Wang *et al.* [20] devised a federated learning for privacy protection system under edge computing. First, a weight mask and shared secret protocol are created based on a random mask method of secret transferring, and they provide lightweight privacy protection. A hash function and digital signature have been developed to protect the message against replay attacks and to guarantee its consistency and integrity.

In 2022, L. Wang *et al.* [21] devised multi-layer edge computing to recognize hazards in latency and secure ways. Physiological data series are captured and processed using the gated recurrent units (GRU) model, whereas fine-grained detection models are customized to fit scenario-specific data and personal health profiles. The edge computing framework places physiological data storage and detection algorithms close to end users for reduced latency, while encryption protects data privacy.

In 2023, I. Khan *et al.* [22] suggested an authorization and authentication system that protects users' medical information. In order to increase the security of allowed data access in the fog environment and to ensure the confidentiality of medical information, the proposed technique employs user behavior profiling and decoy technology. The suggested model effectively secures medical data. However, it may face challenges in scalability and managing the complexity of decoy deployment.

In 2023, L. Meng and D. Li [23] presented an edge-based privacy preserving approach (EBPPA) for healthcare applications. EBPPA addresses the issues of healthcare applications by combining the XGBoost algorithm, which allows for accurate computations, with homomorphic encryption, which safeguards medical data. The suggested technique offers improved data privacy, integrity and confidentiality by utilizing virtual edge nodes and secret sharing, which makes it a viable option for protecting private patient information in the healthcare industry.

In 2023, M. Izhar *et al.* [24] devised EEC and edge nodes that guarantee both integrity and privacy. The peer-to-peer DLT approach allows for decentralized data sharing, eliminating the need for centralized servers, while edge nodes provide in-vivo data gathering and monitoring. Ethereum's public edge DLTs encourage data ownership and accessibility, and a hybrid machine learning model improves system performance by anticipating and identifying security risks early.

From the literature survey, key challenges in secure data handling for IoT-enabled healthcare systems emerge. Existing frameworks often struggle with scalability and interoperability, complicating integration across diverse devices. Additionally, computational and resource constraints limit the practicality of security measures, while privacy-preserving methods can introduce latency, hindering timely healthcare delivery. In order to tackle these challenges, this paper proposed a novel in an edge computing environment using deep learning networks.

3. PROPOSED SECURE-HMS METHODOLOGY

In this research, a Secure healthcare monitoring system (Secure-HMS) in an edge computing environment has been proposed. In the data acquisition layer, the healthcare information of a person is collected by the sensors of several devices. The gate-

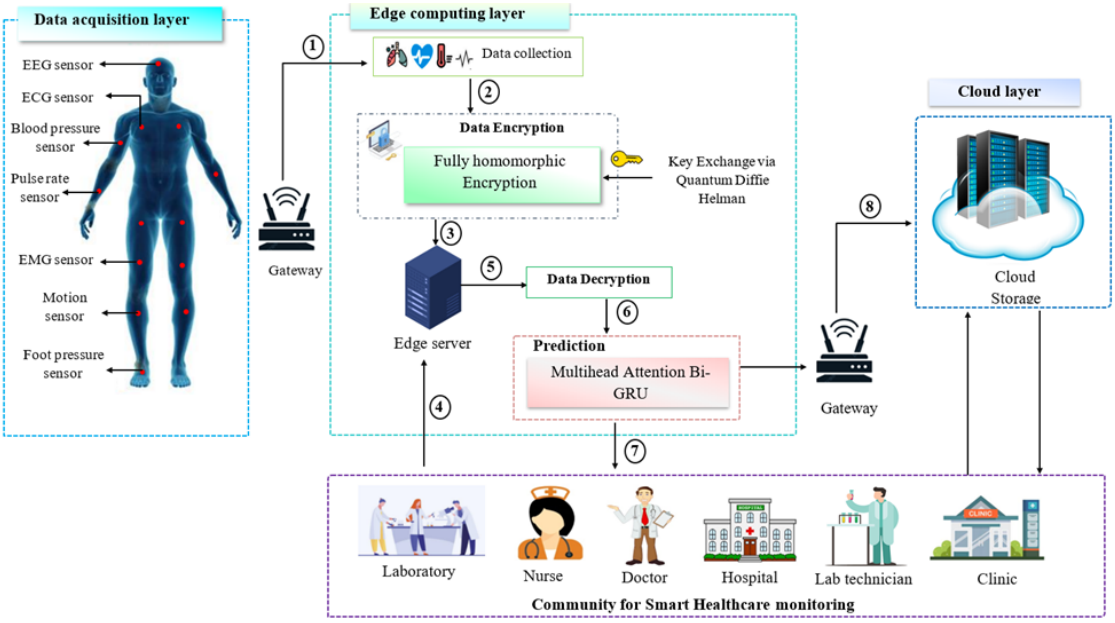


Fig. 1. Proposed Secure-HMS framework

way devices then transmit the data to the edge layer. The edge server is in charge of keeping all sensitive patient data in the cloud at the edge computing level. This edge layer contains an encryption module that employs fully homomorphic quantum Diffie-Hellman encryption to protect patient data security and privacy. During data access, authentication is performed in the edge server. After authentication, the encrypted data will be decrypted and prediction will be performed using multi-head attention-based Bi-GRU for enhancing overall system efficiency. After prediction, the predicted data will be sent to the healthcare community and also stored in cloud storage. Figure 1 illustrates the overall flow diagram of the Secure-HMS framework.

3.1. Data acquisition layer

In the data acquisition layer, a variety of sensors are utilized, including the EEG sensor (electroencephalogram), ECG sensor (electrocardiogram), blood pressure sensor, EMG sensor (electromyography), pulse rate sensor, motion sensor and foot pressure sensor, to capture critical health data for continuous monitoring. The EEG sensor monitors brain activity, providing insights into neurological health, while the ECG sensor records the heart's electrical signals to detect potential cardiovascular issues. A blood pressure sensor measures systolic and diastolic pressure, essential for managing heart-related conditions, and the pulse rate sensor tracks the heart beats per minute for real-time heart function monitoring. The EMG sensor detects muscle activity, useful for evaluating neuromuscular function, and the motion sensor tracks body movement, helping monitor mobility and detect falls. Lastly, the foot pressure sensor assesses pressure distribution on the feet, which is critical for detecting gait abnormalities or conditions like diabetic foot ulcers. These sensors collectively transmit data through a gateway for processing at the edge, ensuring real-time health monitoring and secure data handling.

3.2. Data encryption

Fully homomorphic encryption is a cryptographic technology that allows encrypted data to be processed without decryption. Typically, FH operations enable unlimited calculations on data, including both multiplication and addition on encrypted data. Herein the Gentry-Sahai-Waters homomorphic encryption is utilized for encrypting the data. The GSW13 scheme enables secure data processing by encoding patient data in ciphertext, allowing for computations on encrypted information while preserving privacy. Let N be the number of layers in the relatively homomorphic system that is appropriate for safely processing patient data, and let λ be a security parameter. Below are the primary steps adapted for handling patient health information securely, focusing on the relevant aspects for preventing adaptive key recovery attacks.

3.2.1. Parameter setup

Select modulus g of $t = t(\lambda, N)$ bits, error distribution $e = e(\lambda, N)$, and lattice dimension parameter $d = d(\lambda, N)$ for LWE that achieves at least 2^{λ} security against known attacks.

Select parameter $p = p(\lambda, N) = 0\,(dlog(g))$; and the output parameters are represented as $params = (d, g, e, p)$.

3.2.2. Key generation

Generate secret vector $s \in k_g^e$ and compute the derived vector s as follows:

$$v = \begin{bmatrix} 1 \\ -s \end{bmatrix} \in k_g^{e+1}. \tag{1}$$

Construct public matrix x using random matrix $y \in k_g^{p*e}$ and error vector $u \in z^p$:

$$y = Y.S + u \in k_g^p, \tag{2}$$

where A is formed by concatenating b and the columns of B, ensuring:

$$X.v = u. \tag{3}$$

This setup yields public key $pk = x$ and private key $sk = s$.

3.2.3. Encryption of patient data

Encrypt a patient data bit $\mu \in \{0,1\}$ by generating random binary matrix B and computing ciphertext C:

$$C = Flatten\,(\mu \cdot I_N + BitDecomp(B \cdot x)) \in k_g^{d*d}. \tag{4}$$

Here I_N is the identity matrix, ensuring that data can be securely processed without revealing μ.

3.2.4. Decryption

Recover the message μ by calculating inner products using private key $sk = s$

$$ai = \langle Ci, v \rangle \tag{5}$$

and the final decrypted message:

$$m = \frac{a_i}{b_i}. \tag{6}$$

The GSW13 scheme thus provides a robust homomorphic encryption approach for managing patient data, enabling secure computation on encrypted values.

3.2.5. Quantum Diffie-Hellman key exchange

For exchanging the key from FHR, a revolutionary physics-based quantum DH protocol has been utilized. A quantum state in the QDH protocol will be the equivalent of the final key in the traditional DH protocol. Let's designate the two communication parties, D (healthcare provider) and M (monitoring device), which can exchange qubits. Party D (the healthcare provider) and party M (the monitoring device) need to share initial values: a set of private quantum states represented by secret values t_D for D and t_m for M. These values help establish secure, ephemeral states during the key exchange. The protocol also requires a shared, public state TS, set to a common initial state (e.g. $|\mathbf{0}\rangle\mathbf{n}$ that both parties can prepare as a starting point for communication.

1. Party D creates qubit state |0i using the public information provided in TS (representing the initial state for secure data communication).
2. D then uses its secret t_D to apply a unitary transformation $X(t_D)$ resulting in the state of $\left||D_m = X(t_D)|0_n\right)$ This qubit is then transmitted to M over a quantum channel.
3. Party M receives $|D_n$ and alters it using its unitary operator $X(tM)$ to create the state of $\left|MZ_n = X(tM)\right|P_nM$.
4. Following the protocol, M then prepares its own initial qubit $|0_n$ using the shared TS, and applies $X(t_m)$ to create qubit $\left|M_n = X(X_m)\right|0_n$. This qubit is then sent back to P.
5. Party P receives $\left|\psi M_n\right.$ and further applies the operations $X(Td)$ and $X(TS)$ to achieve the final state $\left|DM_n := X(TS)X(TX)\right|\psi M_n$. At this point, both parties have achieved identical states: $|\psi dm_n$ and $|\psi md_n$ even without directly exchanging their secret values t_d and t_m.

Figure 2 depicts the key exchange principle of QDH. The QDH protocol offers a high degree of security against eavesdropping since it does not need parties to send the actual secret states. Data from healthcare monitoring can now be encrypted using the derived shared state as a secure session key, protecting patient privacy. Healthcare data frequently contain sensitive patient information; thus, the non-classical characteristics of quantum key distribution assist guarantee confidentiality even in the face of potential network threats.

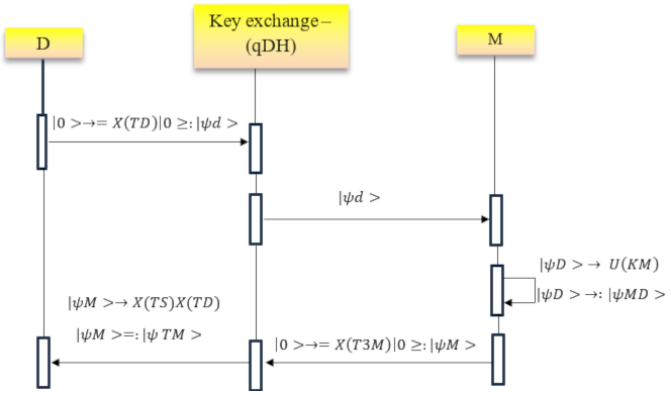


Fig. 2. Key exchange operation

3.3. Prediction via multi-head attention Bi-GRU

Once the data are decrypted, the prediction will be performed using multi-head attention based Bi-GRU for enhancing overall system efficiency. Each neuron contains two gated recurrent units with opposing training directions, which together make up a bidirectional gated recurrent unit (Bi-GRU). Following that, GRU produces a vector with fixed dimensions after calculating the passing word vectors. Here is the detailed design. The architecture of MA-BiGRU is demonstrated in Fig. 3.

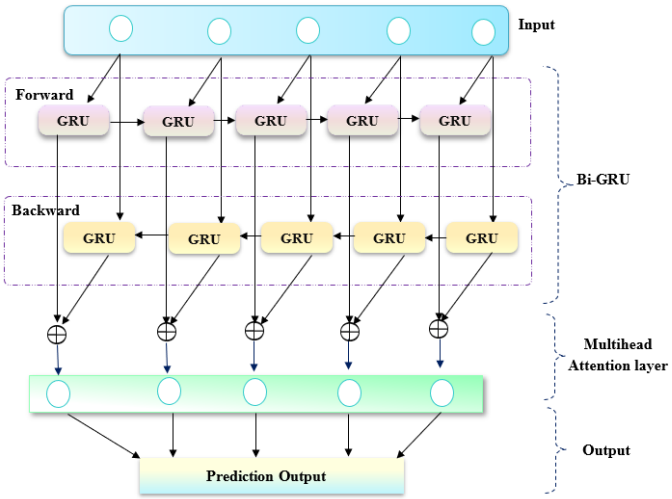


Fig. 3. Architecture of MA-BiGRU

The weighted sum of the forward hidden layer's state $\overrightarrow{f_i}$ and $\overleftarrow{f_i}$ represent the reverse hidden layer which is the output for the n-th instant. The following formulas are used for calculations:

For the n-th instant, the output is the weighted sum of the forward hidden layer's state $\overrightarrow{f_i}$ and the $\overleftarrow{f_i}$ representing the reverse hidden layer. The following formulas are used for calculations:

$$\overrightarrow{f_i} = \mathrm{GRU}\left(v_i, \overrightarrow{f_{i-1}i}\right), \tag{7}$$

$$\overleftarrow{f_i} = \mathrm{GRU}\left(v_i, \overleftarrow{f_{i-1}}\right), \tag{8}$$

$$v_i = x_i \overrightarrow{f_i} + y_i \overleftarrow{f_i} + q_i\,. \tag{9}$$

Four computational components go into GRU. To decide which data to discard earlier, GRU first utilizes a reset gate.

$$t_i = \sigma\left(a_m S_n + t_m f_{i-1} + q_t\right). \tag{10}$$

Next, GRU uses the update gate to update the data that are currently available.

$$x_n = \sigma\left(p_m S_n + t_m k_{n-1} + d_m\right). \tag{11}$$

GRU then calculates the candidate memory content, where d is the bias and p and t are the weight information, to ascertain the output of the current instant:

$$\widetilde{k}_n = \tan\left(p_k S_n + t_k v_n \odot k_{n-1} + d_m\right). \tag{12}$$

Lastly, the output from the aforementioned outcomes is computed by GRU:

$$\mathrm{GRU}\left(S_n, k_{n-1}\right) = (1 - v_n) \odot k_{n-1} + v_n \widetilde{k}_n\,. \tag{13}$$

S_n represent the input value of the network at the initial instant. k_{n-1} denote the hidden layer at the last moment. $\odot$ denotes the Hadamard product. σ denotes the sigmoid function. Thus, enabling the GRU network to learn discriminative features from the input features and to improve the performance.

3.3.1. Multi-head attention layer

The attention mechanism enables the crucial extraction of pertinent information to be recorded. The attention function links a collection of key-value pairs to a query (q). The assessment of the attention mechanism is divided into three phases. To begin, a similarity function $s(q, w_i)$ is produced to calculate the similarity between f and each w. Equation (14) shows how dot, general and concat have numerous similarity functions that lead to a comparable attention score (c):

$$s(q, w_i) = \begin{cases} q^D w_i & \text{Dot,} \\ q^D k w_i & \text{General,} \\ k\left[q; w_i\right] & \text{Concat.} \end{cases} \tag{14}$$

Then, use a softmax function to get the weighted vector β. The context vector c is ultimately obtained by adding the weighted values of β and E:

$$\beta_i = \mathrm{softm}(s_i), \tag{15}$$

$$c = \sum_{d=1}^{m} \beta_i z_i\,. \tag{16}$$

Queries (q) and keys (w) have dimensions of n_l, while values (d) have dimensions of x_d. The scaled dot-product attention approach determines attention scores using the following formula:

$$\mathrm{Atten}(s, f, w) = \mathrm{Softmax}\left(\frac{q w^D}{\sqrt{x_l}}\right) d. \tag{17}$$

A multi-head attention mechanism is composed of several attention layers that collaborate to enable the architecture to attend to data from different visual subspaces in various regions simultaneously.

$$\mathrm{head}_i = \mathrm{Atten}\left(\left(q k_i^q, w q_i^w, d q_i^d\right)\right), \tag{18}$$

$$\mathrm{MultiHead}(q, w, d) = \mathrm{Conca}\left(\mathrm{head}_1, \cdots, \mathrm{head}_m\right) q. \tag{19}$$

Each of these operations is referred to as a "head", and after q repetitions, the vectors generated by the parallel heads are combined to form a single vector.

The healthcare dataset contains 2000 samples. The split consisted of 70% data training and 30% data testing to ensure an effective model evaluation. To gain the edge of generalization and at least minimize the risks of overfitting, the model was trained for 100 epochs using a batch size of 32. The Adam optimizer was applied with an initial learning rate of 0.001. Early stopping was utilized with monitoring of the validation loss so that the model could stop training once performance was satisfactory and while the onset of unnecessary training could still be prevented. The detailed training parameters are presented in Table 1.

Table 1
MA-BiGRU model training parameters

Training parameter	Value
Dataset split	70% training, 30% testing
Learning rate	0.001
Batch size	32
Validation strategy	Early stopping (validation loss)
Optimizer	Adam
Number of epochs	100

4. RESULT AND DISCUSSION

This section assesses the efficiency of Secure-HMS for safe healthcare systems using computer-based simulations. The experimental setup, simulation scenarios and performance evaluation metrics are described, followed by an analysis of the

simulation outcomes with different parameters. MATLAB was used to assess the Secure-HMS system under various scenarios. The authentication mechanism and edge server have been simulated using separate threads.

4.1. Dataset description

The experiment yields values for the vital parameters (attributes) of an average person. The selected characteristics are basic user body parameters used for health monitoring. The sample size is 250 KB, and a set of 1500 sample data are aggregated. Every 2000 sample data points are collected in 3.5 seconds using the experimental setup described above. The user's private information is eliminated from the raw data before sampling. The most prevalent characteristics of healthcare data are age and gender, with other crucial attributes including heart rate, blood pressure, oxygen saturation and body temperature. An individual's body temperature ranges from 36.1°C to 37.2°C. The typical ranges for resting pulse rates are 12–16 breaths per minute and 60–100 beats per minute. A grown person's normal foot pressure is between 100 and 120 pounds per square inch (psi).

4.2. Performance of the proposed Secure-HMS model based on edge computing

The experiment was conducted with and without the edge computing layer. The cloud side handles all of the data processing for the 2000 collected samples since there is no edge computing layer. The sample is reduced by 30% when using the edge computing layer. The edge gateway processes 70% of the sample data. Table 1 displays the outcomes of the processed sample size reduction. Table 2 displays the energy consumption and transfer time in the edge node during the 250 Kb sample collecting and processing. Transmission time and energy usage without an edge gateway are also displayed in Table 3. The outcome of the trial showed that the proposed framework uses 68.56% less energy and cuts the transfer time by 63.12%.

Table 2
Processed sample size

	Number of samples	With edge computing	Without edge computing
Value reduction (%)	2000	Data processed at the edge node	Data processed at the cloud
		1400	600
		70	30

4.3. Performance evaluation

In this section, efficiency analysis is performed via the proposed Secure-HMS with various existing approaches based on encryption and decryption time, security analysis control overhead, average delay, average response time, PDR and throughput. A comparison is made between the Secure-HMS model and existing methods such as the MEdge-chain [18], SDN-based edge network [16] and EBPPA [23]. Table 4 demonstrates the GSW13 homomorphic encryption processing time for sample healthcare data.

Table 4
GSW13 homomorphic encryption processing time for sample healthcare data

Data type	Sample value	Encryption time (ms)	Decryption time (ms)	Ciphertext size (KB)
Heart rate (bpm)	75	22.1	18.7	4.5
Blood pressure (mmHg)	120/80	24.3	19.2	5.1
Oxygen saturation (%)	98	21.8	18.5	4.4

Table 4 demonstrates the encryption time, decryption time and size of the ciphertext for three sample healthcare parameters that could be processed using GSW13 homomorphic encryption in the proposed Secure-HMS framework. Heart rate, blood pressure and oxygen saturation have been reported, resulting in encryption times of 21.8 ms to 24.3 ms and decryption times of 18.5 ms to 19.2 ms. The size of ciphertexts is between 4.4 KB and 5.1 KB. These results prove that the proposed scheme provides secure data processing (preserving data confidentiality) with low computation time and small size of ciphertext, suitable for real-time healthcare monitoring scenarios.

Figure 4 shows the comparison of encryption and decryption time used with the Secure-HMS middleware framework for three typical healthcare data types, namely blood pressure, heart rate and oxygen saturation with GSW13 homomorphic encryption. It is noted from the results that encryption time falls between 21.8 ms and 24.3 ms, and decryption time lies between 18.5 ms and 19.2 ms for various data types. These results prove that the Secure-HMS proposal does not compromise the efficiency of encryption and decryption and thereby enables secure processing of private healthcare data with low computational overhead, which is a requirement for real-time healthcare monitoring applications.

Table 3
Reduction of transmission time and energy usage with edge gateway

Metrics devices	Transmission time			Reduction	Energy usage			Reduction
	Without edge	Sensor to edge gateway	Edge gateway to cloud		Without edge	Sensor to edge gateway	Edge gateway to cloud	
Value	1785	234.12 ms	120.09 ms	63.15%	936.4 MJ	120.12 MJ	38 MJ	68.56%

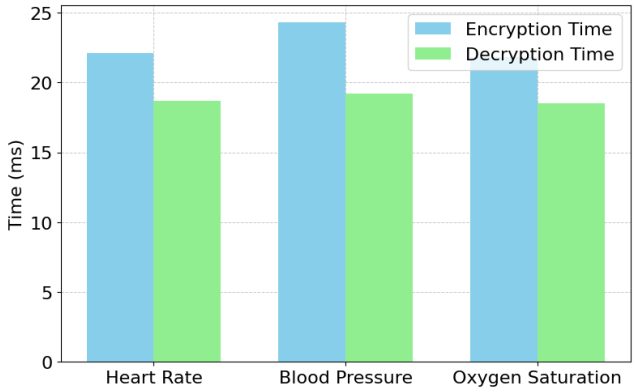


Fig. 4. Encryption and decryption time comparison

Encryption and decryption time

The time needed to encrypt the given data is known as encryption time, while the time needed to obtain the encrypted data is known as decryption time. Both times are measured in milliseconds.

Figure 5 demonstrates the comparison of decryption and encryption time with proposed and existing techniques. It clearly shows that Secure-HMS has the lowest encryption time and decryption time, indicating its efficiency, while MEdge-chain exhibits the highest times for both encryption and decryption. Overall, this visualization highlights the effectiveness of each method, emphasizing the proposed approach's advantages in speed.

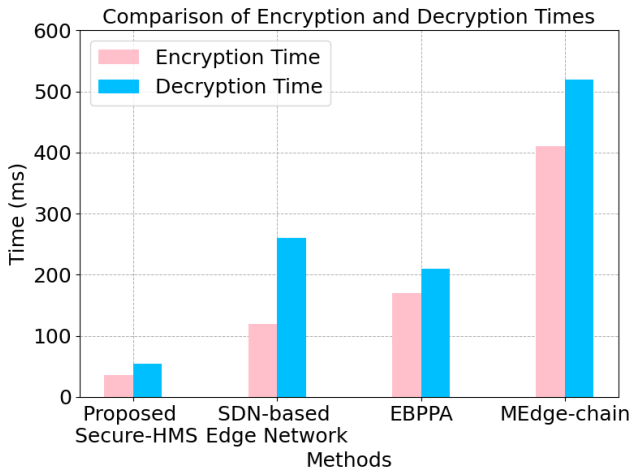


Fig. 5. Decryption and encryption time comparison

Security analysis

Security analysis refers to evaluating how effectively the edge server safeguards patient data from unauthorized access, breaches or tampering during processing and transmission. This includes assessing the server's encryption, authentication protocols and the ability to resist threats, ensuring data privacy and integrity for healthcare users.

Figure 6 demonstrates the comparison of security performance based on Secure-HMS vs current techniques. Proposed Secure-HMS offers the highest security with strong encryption and threat resistance. SDN-based edge network ensures secure data transmission through dynamic traffic management. EBPPA and MEdge-chain provide good security but may have slight limitations in privacy and threat resistance as compared to the other methods. The proposed Secure-HMS improves the range of overall security rate by 4.08%, 3.06%, and 2.04% in comparison with MEdge-chain, EBPPA and SDN-based edge network, respectively.

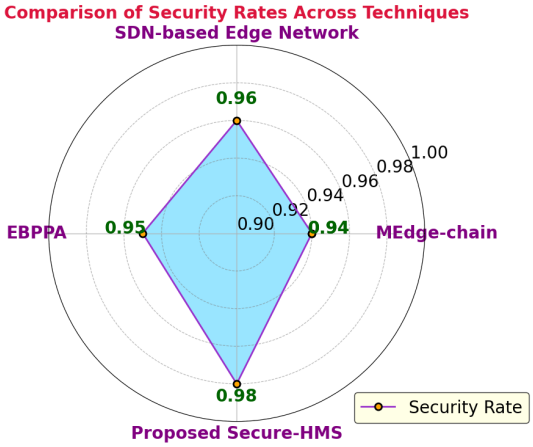


Fig. 6. Security analysis comparison

Average response time

Average response time refers to the duration required by the edge server to finish processing data and deliver the outcomes to patients.

The total execution time, comprising the waiting time for each edge server, is displayed in Fig. 7. Response time will be minimal in an effective edge computing system. The time required to finish them rises as the number of servers rises. The proposed Secure-HMS method has the fastest completion time when compared to other existing techniques.

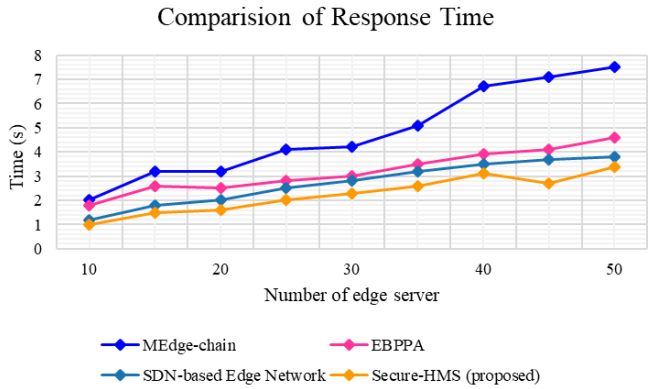


Fig. 7. Response time comparison

Packet delivery ratio

The number of packets that come in compared to the number that are sent out is called PDR.

Figure 8 demonstrates the comparison of PDR with the Secure-HMS model with the existing techniques. Compared to the current methods, the suggested approach delivers a greater packet delivery ratio.

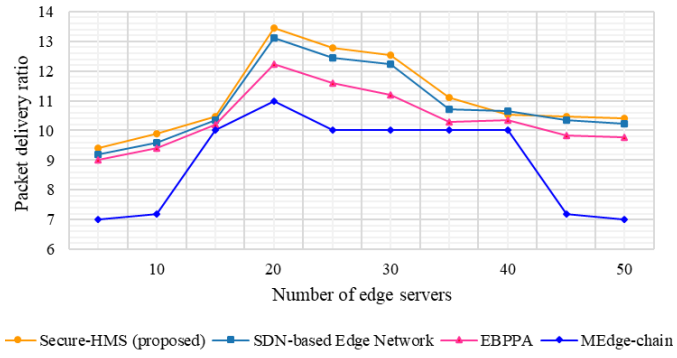


Fig. 8. Comparison of PDR

Average delay

Delay (δ) refers to the complete time required for a packet to reach its destination successfully, while the average delay is calculated by summing all delay measurements and dividing by their count.

Figure 9 demonstrates the comparison of average delay of the edge server. The existing techniques fail to effectively minimize delay because of the time necessary to set up the initial path. The proposed work increases 0.34 s of delay even with 50 edge servers. Therefore, the proposed Secure-HMS method reduces the delay efficiently.

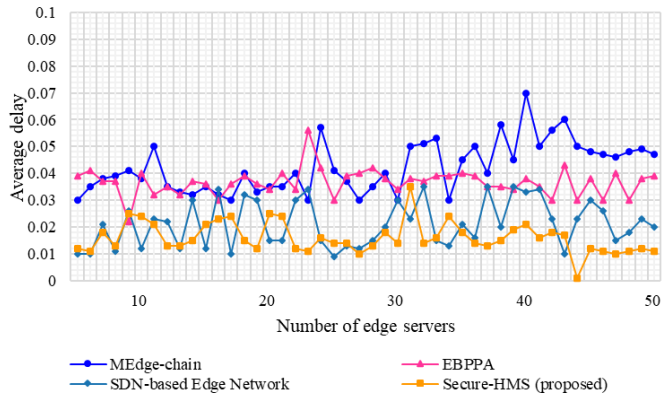


Fig. 9. Average delay comparison

Throughput (η)

Network throughput is calculated as the sum of the data transfer rates received by each node in a network.

The throughput of the proposed vs existing techniques is shown in Fig. 10. The proposed Secure-HMS has higher throughput because of its intelligent decision-making, edge communication and efficient utilization of network resources. IoT devices in other networks are unable to effectively process massive amounts of data; hence, the proposed network generates better throughput than other networks.

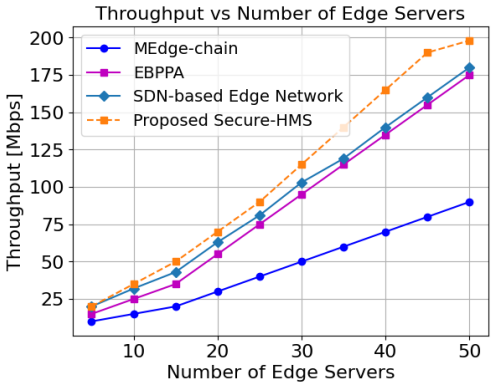


Fig. 10. Throughput comparison

Control overhead

The ratio of the total number of control messages produced by each node in a network to the number of packets that are successfully received is known as control overhead.

Figure 11 displays the comparison of control overhead. The MEdge-chain scenario has smaller control overhead because there are fewer control message exchanges. EBPPA, on the other hand, exchanges additional control messages, resulting in larger control overhead. The additional control messages for network optimization in the proposed Secure-HMS network result in relatively greater overhead as compared to the SDN-based edge network.

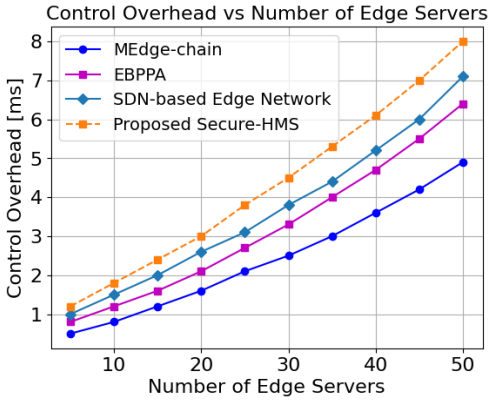


Fig. 11. Comparison of control overhead

5. CONCLUSIONS

In this research, the Secure-HMS framework integrates edge computing and deep learning networks, delivering a secure and efficient healthcare monitoring system. With fully homomorphic quantum Diffie-Hellman encryption, it ensures high levels of data integrity and confidentiality, while the multi-head attention-based Bi-GRU model significantly enhances prediction accuracy and system efficiency. The experimental results of the proposed framework demonstrate notable advantages, including enhanced data security and confidentiality, as well as improved prediction accuracy for healthcare monitoring. The proposed framework is effective for real-time applications because it decreases latency and speeds up response times by processing

data at the edge. Additionally, the Secure-HMS framework's robust encryption and authentication mechanisms safeguard sensitive patient information, offering a comprehensive solution for secure healthcare data handling. In the future, the proposed Secure-HMS can focus on reducing the computational costs of homomorphic encryption, expanding compatibility to include a wider variety of healthcare data, and further lowering latency to support real-time decision-making. Additionally, incorporating adaptive learning algorithms may enable the system to dynamically respond to evolving healthcare needs and data patterns.

ACKNOWLEDGEMENTS

The author would like to express his heartfelt gratitude to the supervisor for his guidance and unwavering support during this research process.

REFERENCES

[1] A. Singh and K. Chatterjee, "Edge computing based secure health monitoring framework for electronic healthcare system," *Cluster Comput.*, vol. 26, no. 2, pp. 1205–1220, 2023, doi: 10.1007/s10586-022-03717-w.

[2] A.R. Aarthy and K. Vinoth Kanth, "DNA Encryption-Based Secure Access Control And Data Sharing In An Enabled Cloud Environment," *Int. J. Syst. Des. Comput.*, vol. 2, no. 1, pp. 14–19, 2024.

[3] G. Sreetha, and T.V. Chitra, "Block chain assisted cloud based medical data storage Via Quantum Diffie-Hellman Key Exchange," *Int. J. Comput. Eng. Optim.*, vol. 1, no. 1, pp. 01–9, 2023.

[4] D.F. Ojo-Seriki, *Design and implementation of personal health data monitoring and retrieval system for health providers*, M.S. thesis, Vaal Univ. of Technology, South Africa, 2020.

[5] M. Çinar, E. Aslan, and Y. Özüpak, "Comparison and optimization of machine learning methods for fault detection in district heating and cooling systems," *Bull. Pol. Acad. Sci. Tech. Sci.*, vol. 73, no. 3, p. e154063, 2025, doi: 10.24425/bpasts.2025.154063.

[6] A. Jenice Prabhu *et al.*, "Efficient Data Security Using Hybrid RSA-TWOFISH Encryption Technique on Cloud Computing," in *International Conference on Frontiers of Intelligent Computing: Theory and Applications*, pp. 495–502, 2023, doi: 10.1007/978-981-99-6706-3_42.

[7] H.K. Kalidindi, "Crow Search Optimized DNA Encryption for Secure Medical Data Transmission," *Int. J. Comput. Eng. Optim.*, vol. 2, no. 3, pp. 80–85, 2024.

[8] E. Aslan, "Temperature Prediction and Performance Comparison of Permanent Magnet Synchronous Motors Using Different Machine Learning Techniques for Early Failure Detection," *Eksploat. Niezawodn.*, vol. 27, no. 1, pp. 1–16, 2025, doi: 10.17531/ein/192164.

[9] M. Anisha, and V.A. Beenu, "Double secure cloud medical data using Euclidean distance-based Okamoto Uchiyama homomorphic encryption," *Int. J. Syst. Des. Comput.*, vol. 2, no. 01, pp. 1–7, 2024.

[10] F. Alpsalaz, "Fault detection in power transmission lines: Comparison of chirp-Z algorithm and machine learning based prediction models," *Eksploat. Niezawodn.*, vol. 27, no. 4, pp. 1–15, 2025, doi: 10.17531/ein/203949.

[11] D.C. Nguyen, P.N. Pathirana, M. Ding, and A. Seneviratne, "Blockchain and edge computing for decentralized EMRs sharing in federated healthcare," in *GLOBECOM 2020-2020 IEEE Global Communications Conference*, pp. 1–6, 2020, doi: 10.1109/globecom42002.2020.9347951.

[12] R. Surendiran *et al.*, "Modified Elephant Herd Optimization-Based Advanced Encryption Standard," in *International Conference on Frontiers of Intelligent Computing: Theory and Applications*, pp. 519–528, 2023, doi: 10.1007/978-981-99-6706-3_46.

[13] S. Datta, and S. Namasudra, "Blockchain-based smart contract model for securing healthcare transactions by using consumer electronics and mobile edge computing," *IEEE Trans. Consum. Electron.*, vol. 70, no. 1, pp. 4026–4036, 2024, doi: 10.1109/tce.2024.3357115.

[14] A.K. Alnaim, and A.M. Alwakeel, "Machine-learning-based IoT–edge computing healthcare solutions," *Electronics*, vol. 12, no. 4, p. 1027, 2023, doi: 10.3390/electronics12041027.

[15] H. Gao, H. Huang, L. Xue, F. Xiao, and Q. Li, "Blockchain-enabled fine-grained searchable encryption with cloud–edge computing for electronic health records sharing," *IEEE Internet Things J.*, vol. 10, no. 20, pp. 18414–18425, 2023, doi: 10.1109/jiot.2023.3279893.

[16] J. Li *et al.*, "A secured framework for sdn-based edge computing in IOT-enabled healthcare system," *IEEE Access*, vol. 8, pp. 135479–135490, 2020, doi: 10.1109/access.2020.3011503.

[17] M.S. Christo, V.E. Jesi, U. Priyadarsini, V. Anbarasu, H. Venugopal, and M. Karuppiah, "Ensuring improved security in medical data using ecc and blockchain technology with edge devices," *Secur. Commun. Netw.*, vol. 2021, no. 1, p. 6966206, 2021, doi: 10.1155/2021/6966206.

[18] A.A. Abdellatif *et al.*, "Medge-chain: Leveraging edge computing and blockchain for efficient medical data exchange," *IEEE Internet Things J.*, vol. 8, no. 21, pp. 15762–15775, 2021, doi: 10.1109/jiot.2021.3052910.

[19] A. Singh, and K. Chatterjee, "Securing smart healthcare system with edge computing," *Comput. Secur.*, vol. 108, p. 102353, 2021, doi: 10.1016/j.cose.2021.102353.

[20] R. Wang, J. Lai, Z. Zhang, X. Li, P. Vijayakumar, and M. Karuppiah, "Privacy-preserving federated learning for internet of medical things under edge computing," *IEEE J. Biomed. Health Inform.*, vol. 27, no. 2, pp. 854–865, 2022, doi: 10.1109/jbhi.2022.3157725.

[21] L. Wang, B. Xu, H. Cai, and P. Zhang, "Context-aware emergency detection method for edge computing-based healthcare monitoring system," Trans. Emerging Telecommun. Technol., vol. 33, no. 6, p. e4128, 2022, doi: 10.1002/ett.4128.

[22] I. Khan, A. Ghani, S.M. Saqlain, M.U. Ashraf, A. Alzahrani, and D.H. Kim, "Secure Medical Data Against Unauthorized Access Using Decoy Technology in Distributed Edge Computing Networks," *IEEE Access*, vol. 11, pp. 144560–144573, 2023, doi: 10.1109/access.2023.3344168.

[23] L Meng, and D. Li, "Novel Edge Computing-Based Privacy-Preserving Approach for Smart Healthcare Systems in the Internet of Medical Things," *J. Grid Comput.*, vol. 21, no. 4, p. 66, 2023, doi: 10.1007/s10723-023-09695-6.

[24] M. Izhar, S.A. Naqvi, A. Ahmed, S. Abdullah, N. Alturki, and L. Jamel, "Enhancing Healthcare Efficacy Through IoT-Edge Fusion: A Novel Approach for Smart Health Monitoring and Diagnosis," *IEEE Access*, vol. 11, pp. 136456–67, 2023, doi: 10.1109/access.2023.3337092.