Nanotechnology Perceptions
ISSN 1660-6795
www.nano-ntp.com

Smart Health Care Data Analysis and Storage Management in Edge Cloud Computing Based on Deep Learning Techniques

S. Jayasree[1], K. R. Ananthapadmanaban[2]

[1]*Department of Computer Science, SRM Arts and Science College, Kattankulathur*
[2]*Faculty of Science and Humanities, SRM Institute of Science and Technology, Vadapalani Campus*

The growing amount of sensitive patient data presents new issues for the healthcare system today. A Smart Healthcare System (SHS) is required to preserve data and uphold privacy by enhancing the intelligence of the healthcare system. A patient's bodily data is gathered and stored remotely via a number of Internet of Things (IoT) devices, sensors, and mobile nodes found in a SHS. In order to do machine learning (ML), IoT data are usually moved to a cloud or other centralised system for processing and storing; however, this adds to network traffic and creates latency. This research propose novel method in smart healthcare data analysis based on storage management utilizing edge cloud computing with deep learning model. here the healthcare data has been monitored using multi-sensor model with IoT and it is transmitted to edge cloud computing model for storage. Time-constrained information retrieval from stored data and continuous, replication-free indexing are the two methods used to accomplish concurrency. Deep learning is utilised for each storage instance to classify the restrictions for data augmentation and update. Here the data concurrency is carried out based on convolutional recurrent crow searchTrevally adversarial network. Experimental analysis has been carried out in terms of running memory, power consumption, diagnostic accuracy, robustness, RMSE. The suggested method achieved 96% diagnostic accuracy, 89% robustness, 57% RMSE, 58% Power Consumption, 70% running memory.

Keywords: smart healthcare data analysis, storage management, edge cloud computing, deep learning model, adversarial network.

1. Introduction

In the Fourth Industrial Revolution period, medical equipment as well as information technology (IT) convergence technologies are constantly improving. In an environment of smart mobile devices, such convergent methodis continuously expanding from e-health to u-health and then to m-health. Research on "smart healthcare", similar topics, based on technological breakthroughs, is becoming more and more popular [1]. E-health is a medical service that uses IT to connect medical facilities and computer server systems to enhance delivery of health as well as medical information. U-health provides customers (individuals, medical institutions, and businesses) with health, medical data, knowledge, services, products by integrating IT technology into the current medical systems. Furthermore, it's a medical service that lets users monitor their health at any time, from any location. A medical service known as "m-health" makes use of mobile devices for both public health and medical purposes. The term "smart healthcare" describes a full range of medical services where IT and health-related services come together to offer platforms, devices, systems, and information about healthcare and personal health [2]. Internet of things (IoT) sensors have been used in everyday life and in professional healthcare domains, including remote medical treatment, in recent years. Development of medical methods for communication as well as patient management is facilitated by IoT. Reduction in expenses, better treatment outcomes, better disease management, fewer errors, better patient conditions, better medication administration are some of main benefits of implementing IoT in medical facilities [3]. Current economy has witnessed a technological transformation IoT as well as Edge computing paradigms, which have also made it possible to construct several intelligent healthcare services in an organised and Internet-based manner. The treatment procedure, cost of healthcare services, quality of care can all be enhanced by integrating these technologies into the Smart Healthcare System (SHS) [4]. A massive amount of patient data was produced by IoT devices, exchanged amongst healthcare experts for therapeutic purposes, and saved on the storage server. One major obstacle to the creation of real-time reaction applications is the processing of these data. The e-healthcare system required certain architectural solutions in order to preserve its features, low latency, network bandwidth, security, and services at the same time. These solutions include mobile edge computing, cloud-sea computing, cloudlet, microdata centres, fog, cloud, and cloud computing. At the centre of all of these is the innovation in medical technology known as SHM, which is used to monitor patient conditions, particularly those of those in coma. It does this by utilising sensors and IoT devices such as eye blinkers, temperature monitors, and accelerometers, which measure body movements. Elderly and chronically ill individuals can benefit from these remote monitoring devices [5].In the last few decades, the number of elderly and chronic patients who need remote-control health monitoring systems has increased dramatically. Worldwide rates of hospitalisation and patient care are rising as the number of patients increases. A poll indicates that the US's mortality rate is rising daily, amounting to 770,000 deaths annually. This is the result of mishandled drugs, improper dosage, postponed treatment, etc. In these situations, SHM models are helpful and lessen the workload for personnel and other medical specialists. Stakeholders in a smart healthcare system include patients, physicians, clinicians, and other organisations. The primary contributions are the utilisation of edge nodes to reduce network traffic and latencies for machine learning (ML) jobs, as well as the assessment of the amount of data reduction achievable on the edge without materially affecting the accuracy of ML tasks. By serving as a bridge between IoT devices as well as cloud, edge nodes cut down on the amount of data that is transferred there. To build data encodings that are delivered to the cloud, the edge uses the encoder portion of the trained AE. Cloud-based machine learning tasks can be executed in two ways: either directly using encoded data, or by first restoring the original data using the AE's decoder component before using it for the ML work [6].

2. Related works:

In recent years, research on fog computing has been presented for the analysis of massive data from IoT. Two significant application domains where reliable and real-time results are required are healthcare and remote monitoring. To further enhance this subject, IoT and fog computing have been established. [7] outlines a method for utilizing lot of big data from IoT devices to manage and track patients' health. They assert that a number of technologies could reduce the overall cost of controlling or avoiding chronic illnesses. These devices include ones that show real-time health information, self-medication tools, and ones that continuously monitor health indicators. As more individuals have access to cellphones and high-speed Internet, more people are starting to use mobile apps to manage various health conditions.This study looks into the application of big data in healthcare. In order to tackle the challenges associated with providing home-based healthcare monitoring, the Smart Healthcare Monitoring System, also known as SW-SHMS, was presented in [8]. Through ongoing health monitoring, SW-SHMS can significantly contribute to the comfort of older and disabled individuals, enabling them to live independently in an emergency or other urgent healthcare condition. Thus, SW-SHMS gathers physiological data from patients utilizing wearable sensors as well as sends it to cloud for processing and analysis.A decision support system and home monitoring system are explained in [9] to help doctors diagnose, treat, prescribe, rehabilitate, and track the advancement of patients with Parkinson's disease remotely. This is an expert system that gathers information on tremors and helps medical professionals identify and manage them. The evolution of electrophysiology's function in occupational health was examined in Work [10]. They outlined the advantages of developing wearable as well as smart devices for cardiovascular monitoring and their potential uses, showcasing patterns of mobile ECG device use in various demographics and environmental contexts. In addition to analysing earlier research on traditional ECG devices to pinpoint their shortcomings, author [11] offers some perspectives on how the Internet of Things can advance medical uses for ECG devices. In order to construct a method for ECG extraction as well as analysis utilising mobile computing, Work [12] combines the fields of mobile cloud computing as well as health informatics. They intend to implement the method in a commercial setting. It's common practice to classify and analyse medical photos using deep learning. CNN was utilised by the author [13] to identify handwritten images and assist with early diagnosis of Parkinson's disease. Their algorithm picks up features from signals sent by smart pen, which records handwriting movements during private exams using sensors. The effectiveness of LSTM in the analysis and identification of multivariate time series patterns of medical measures in ICU was investigated in work [14]. In the context of cloud brokering architecture, the optimal location of virtual machines is suggested in [15]. A thorough architecture for a cloud service broker was provided in that study. The main contribution of this research was how a cloud scheduler managed the virtual resources of cloud brokers properly[16].In order to provide required healthcare services to a cloud broker, the authors of this research suggested an optimisation method based on QoS. Objective function of the suggested optimisation method is designed without taking the cloud service provider's service charge into account. In a multi-cloud context, how processing clusters are arranged with cost as the primary consideration is explained in [17]. The writers assessed the method taking into account both cost and point efficiency. Their work was mostly used to many-task computing (MTC) applications. The authors of [18] used the SSS algorithm to do efficient picture analysis. More specifically, the concept offers the capability of utilising remote cloud services for data visualisation.To protect privacy and confidentiality in this instance, a combination of volume rendering techniques and SSS technique is used. Actually, one can process produced portions in their encrypted format using this approach. A novel technological approach that makes online image processing possible using cloud computing is proposed in Work [19]. Primary goal of SSS method is to keep private information out of the hands of cloud providers. In fact, this approach consists of an encryption tool that sends a query to several nodes, each of which tries to process a portion of it. Customers can outsource image processing to off-site servers in this way and keep their user data private.Two UCI Cleveland Heart Disease datasets were examined in [20], and an Android cloud-based software for heart disease prediction was suggested. After comparing Naïve Bayes, Support Vector Machines (SVM), Random Forest, Artificial Neural Networks (ANN), Simple Logistic, they determined that SVM, with its 97.53% accuracy, was the best model. Using IoT, [21] has created a diagnostic algorithm for forecasting Chronic Kidney Disease (CKD). In an effort to support doctors and medical teams in underdeveloped nations, this model was designed to identify early signs of chronic kidney disease. Utilizing Decision Trees, SVM, NN, NB, they were able to reach 97% accuracy, 99% sensitivity, 95% specificity across three separate datasets; Decision Trees yielded the best results. Using the Madelon dataset, the authors of [22] examined the calculation durations required by ML methods Support Vector Classifier (SVC), least-angle regression method (LARS), Elastic Net, KNN, PCA, k-Means on a number of Python libraries. They took into account and contrasted the training times of each model. In [23], authors proposed a HealthFog system that satisfies accuracy, latency, network bandwidth, temporal QoS standards while offering precise predictions about patients with heart conditions through the use of edge and fog technologies. Fog computing used patient's data from a Raspberry Pi that was programmed to provide an accurate prediction, while edge computing brought system closer to user, reducing latency as well as energy usage.

3. Multi-sensor model with IoT based smart health monitoring:

IoT technology is used to construct a wide range of smart health applications, fulfilling the goals of Health 4.0. Figure 1 presents a generic structure for Health 4.0 that shows integrated methods as well as key elements of healthcare architecture. Creating excellent services to help people with various types of medical requirements is one of the most important objectives. To do this, system performance must be increased, resources must be used efficiently, and tools must be optimised. Automated processes and intelligence can speed up routine and simple jobs and improve the accuracy of the outcomes. Furthermore, continuous monitoring and prompt medical intervention can be facilitated via remote access and real-time replies. Lastly, improved diagnosis and more individualised treatment can result from the creation of suitable databases containing comprehensive and readily available medical records.

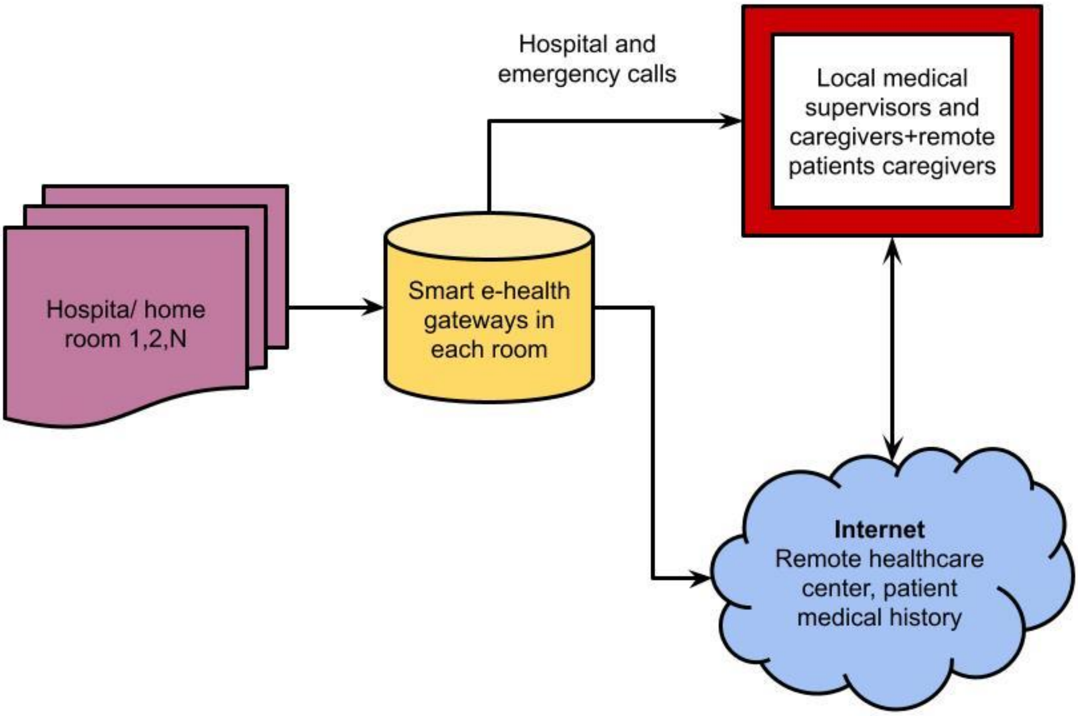
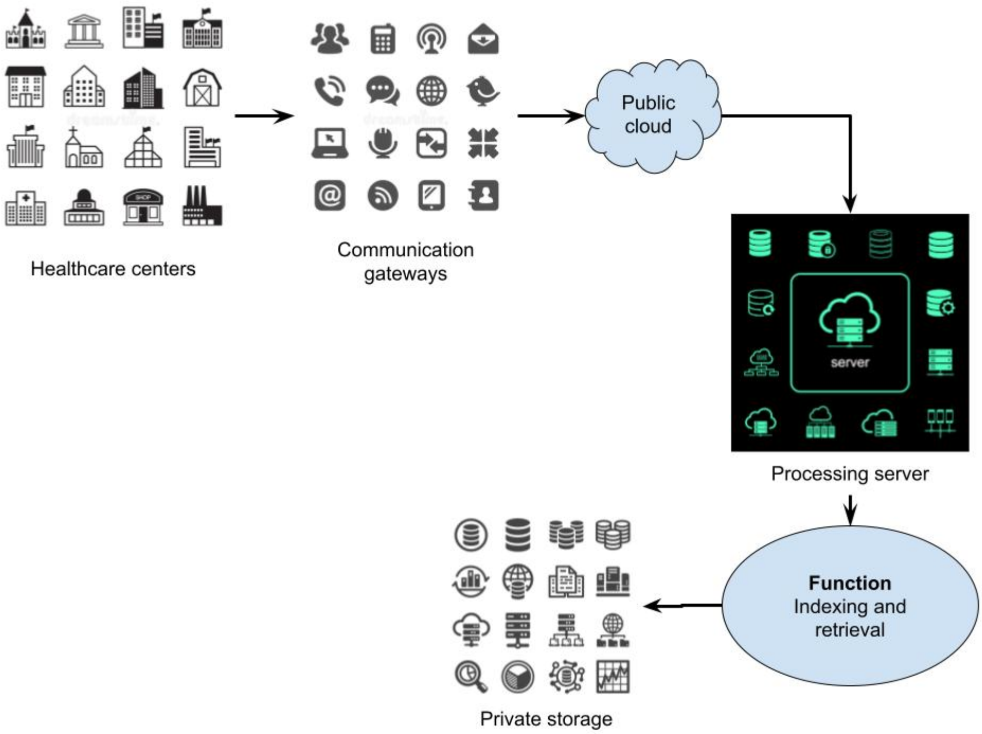


Figure-1 IoT based multi-sensor analysis in smart health monitoring

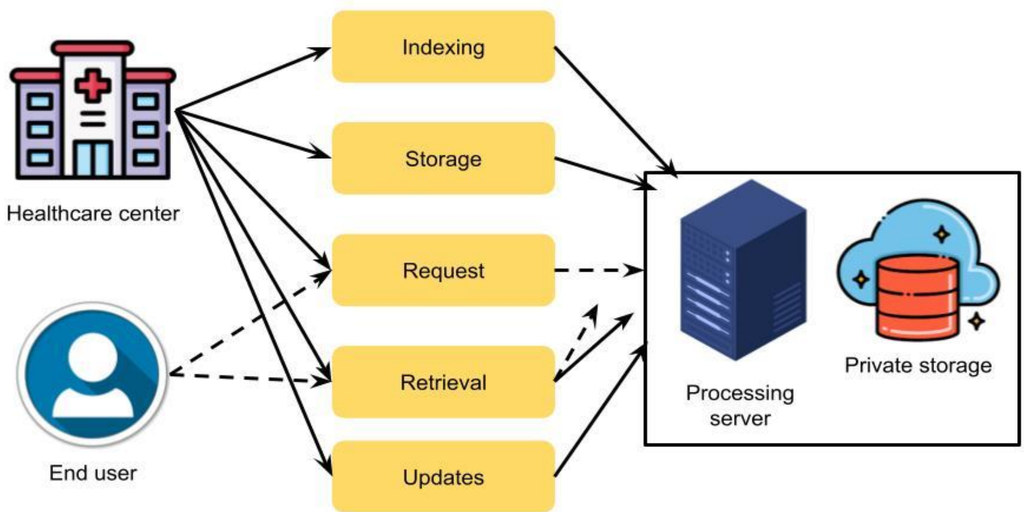
Enhancing operational effectiveness at the same time as preserving low prices, low resource usage, low energy consumption is another equally important goal. Throughput as well as resource consumption must, however, be balanced, and this relies on the system's requirements. Achieving the most performance throughput while using the fewest resources is the best strategy. Regardless of its ultimate optimisations, the use of IoT can help with disease prediction, diagnosis, and health monitoring the vast amounts of data that health sensors gather. The diagnosis process is facilitated and accuracy increased by the instantaneous transmission, analysis, storage of this data to cloud services. Because the expensive techniques of health assessment are replaced by affordable, easily available, user-friendly, instantly responsive alternatives, there will be less strain on healthcare staff and resources. Lastly, there is easy and quick data sharing and collaboration between various healthcare facilities and providers.

Data storage based on edge cloud computing model:

Enhancing concurrency of medical data storage as well as retrieval in healthcare-based clouds is design objective of the suggested system. Electronic health records, or EHRs, are used to store medical data from various hospitals and diagnosis centres. This makes the data easier to share, retrieve, and access. The medical facilities have access to private cloud storage that they pay for in order to carry out the aforementioned tasks. Nonetheless, level-dependent access must be taken into account while maintaining a private cloud due to concurrency, accessibility, and delay-less sharing. Level-dependent access divides authority as well as control among diagnosis centre users in order to protect their privacy and security. Healthcare cloud computing infrastructure and its access functions are depicted in Figures 2(a) and 2(b).



a



b

Figure 2 (a) Cloud based health computing environment, (b) Access Functions

Healthcare centres store EHR in designated private storage that has indexing as well as retrieval capabilities, as shown in Figure 2(a). Concurrent operations between healthcare centres and end users are handled by the processing server. Figure 2 (b) illustrates the differences in the end-users' and healthcare center's activities. While end users are in charge of request and retrieval activities, the healthcare centre is in charge of indexing, storing, retrieving, and updating EHR. A range of health gadgets, including blood pressure monitors, smart watches, wristbands, and electrocardiographs, are part of IoT layer that collects health data as well as monitors numerous physiological signs of the human body.IoT devices are unable to process these data effectively due to restrictions in computational power, storage capacity, power constraints in their hardware. To get the results of a health analysis, users must either bring their data to the hospital or upload it to the cloud. Nevertheless, neither of the systems can provide an early warning system for prospective health issues because they are not handy or real-time.

In order to efficiently minimise computer system latency, lower data transmission bandwidth, safeguard data security, privacy, edge layer runs computing operations on resources near the data source.Executing edge AI reasoning, running cloud-distributed AI method, returning execution results to user or cloud are all handled by edge layer. In our architecture, a DL method for processing related data is deployed for diagnosis at an edge device, which receives the monitored data from the IoT monitoring device over a local area network. It should be noted that in order to run effectively at the edge devices, DL method for edge computing must be redesigned to strike a compromise between method complexity as well as diagnosis accuracy. Privacy of user data is protected by the introduction of the edge computing node, which limits data transmission to user-controlled local area networks.

Global, non-real-time, long-term big data processing and analysis is provided by the cloud platform layer. AI method is sent to edge nodes for execution after centralised AI method training based on business needs, historical data, real-time data, feedback from AI execution. Examining how clouds as well as edges interact is crucial for implementing cloud and edge integration. Even while IoT and edge computing devices are sufficient for effective health monitoring and diagnosis, users can still upload a portion of their data to cloud platform for more precise analysis. DL method used on edge device, as previously indicated, trades off accuracy and complexity.

The third-party service is the highest tier of the architecture. Users can grant permission to private physicians or appropriate hospitals to access their own data via a cloud platform in order to receive a more thorough diagnostic. In order to provide patients with one-stop nursing, medical, insurance services, the cloud platform has the intelligence to seamlessly and cooperatively connect patients, medical professionals, medical service providers, insurance companies.

Data exchange between S and CU: S keeps the data that has been filtered and comes from the F. Only S serves as the data provider for the CU and stores the data in this design. The streamlined overview of the data exchange between S and the CU is displayed in Stage 1. Every time S transmits data to CU, S concurrently transmits the MDi to C, and C begins to await the hash from the BC transaction. The hash is sent from BC to C when the data transfer between S and CU is documented on BC as a transaction. Every time the hash is sent from S to CU, a data transaction is finished. The hash is kept in U's account by C. 4) Data exchange between CU and C: The data exchange between CU and C is displayed in Stage 2. The CU receives the raw data from the S in order to determine the outcome.The result of the computation is sent to the C by the CU. MDi is sent to the C whenever CU sends data to the C. C awaits the BC to provide a hash after obtaining the MDi. The BC logs the data transfer from the CU to C as a transaction and forwards the hash to the C. The data transfer from CU to C is finished when C receives the result, MDi, and hash. 5) Information kept at C: During the registration procedure, C keeps track of the U information. C puts all data in the U's account as soon as it receives the MDi and hashes from the N i. U account is identified by the MDi as C has created a device-specific account.

Data analysis using convolutional recurrent crow searchTrevally adversarial network (CRCSTAN):

Figure 3 depicts our model's overall structure. There are 5 convolutional layers and one fully connected layer in this convolutional neural network. We preprocess original data by separating ECG data as well as feeding method with one-dimensional data because duration of ECG signal varies across dataset. After data are batched into CNN network, a 1 × 5 convolution kernel is used to execute a convolution operation on input data, offset value is added.

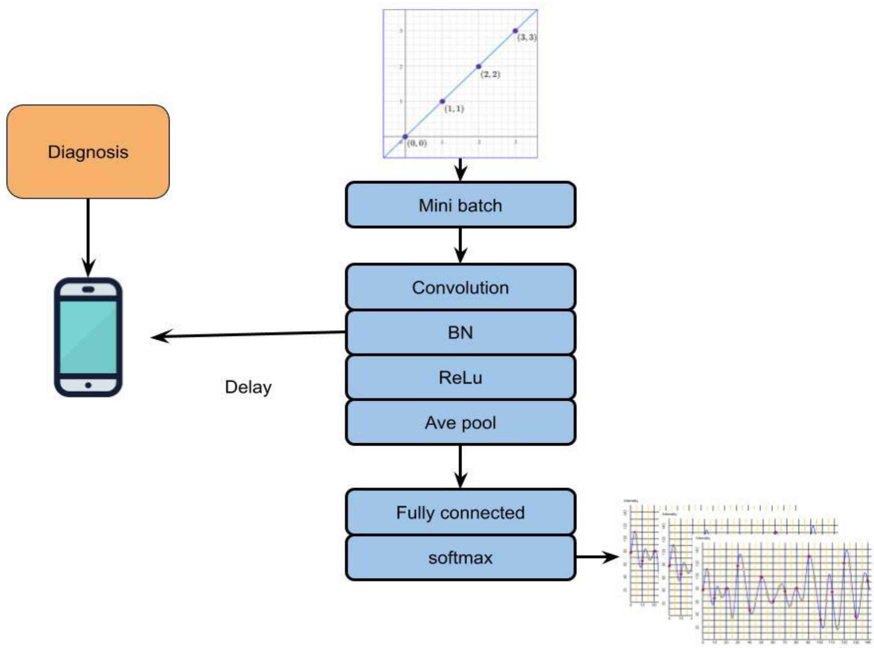


Figure-3 Convolutional architecture based monitored data analysis

We employ batch-normalization (BN) to normalise data after every convolution, which improves network's generalisation and allows the model training process to converge quickly. Due to dispersed distribution of sample features, NN learn slowly or perhaps not at all if the data is not normalised. Equation (1) provides the data normalisation formula:

$$x^{\wedge(k)} = \frac{x^{(k)} - E[x^{(k)}]}{\sqrt{Var\,[x^{(k)}]}} \quad (1)$$

In this case, x (k) denotes the input data's kth dimension, E[x (k)] is average of dimensions, q Var[x (k)] denotes standard deviation. If done too simplistically, layer of expression will be diminished. Therefore, in order to preserve the model's expressiveness, BN adds two parameters (γ and β). Equation (2) is form.

$$y^{(k)} = \gamma^{(k)} x^{\wedge(k)} + \beta^{(k)} \quad (2)$$

Compared to previous activation functions, Rectified Linear Activation (ReLU) can achieve faster convergence, reduce computing complexity, and mitigate overfitting issues to some extent. Equation (3) is the formula for ReLU functions.

$$\text{ReLU}\,(x) = \max(0, x) \quad (3)$$

Its purpose is to set evaluated value to zero if it is less than zero; if not, it stays unchanged. The trained network has been shown to be fully moderately sparse in practice. The training results that are visualised resemble pre-trained outcomes of conventional techniques. The pooling layer can increase the model's capacity for generalisation, avoid overfitting, and lower number of parameters as well as computations while maintaining key features. Method averages two values in the domain using 1 × 2 average pooling. This can lessen the mistake of the rise in the estimate's variance brought on by the neighborhood's constrained size. Output of convolutional layer is transformed into a n × T vector by the fully connected layer. Every number has an associated category (N, A, O, or ~). Most likely prediction result is obtained using softmax. Equation (4) represents the softmax's performance:

$$S_i = \frac{e^{V_i}}{\Sigma_j e^{V_j}} \quad (4)$$

where Vi stands for the array V's i-th element. The prediction label of a sample is determined by taking index of vector with the highest value after it has passed through softmax layer as well as produced a 1 × T vector. Using an activation function based on SoftMax, extracted feature sets are categorised into output illness classes (cout) by eqn (5)

$$c_{out} = \text{SoftMax}\left(\sum_{i=1}^{N_f} f_i * w_i + b\right) \quad (5)$$

Nf is the number of extracted features from equation (8), and f, w, and b stand for the feature values, weights, and corresponding biases. All multidimensional data sets are categorised into scan-specific illness classifications using this method. This procedure is carried out once for every kind of scan and is further enhanced by applying aRNN. This method functions according to Fig. 4 and extracts various feature sets using a LSTM.`

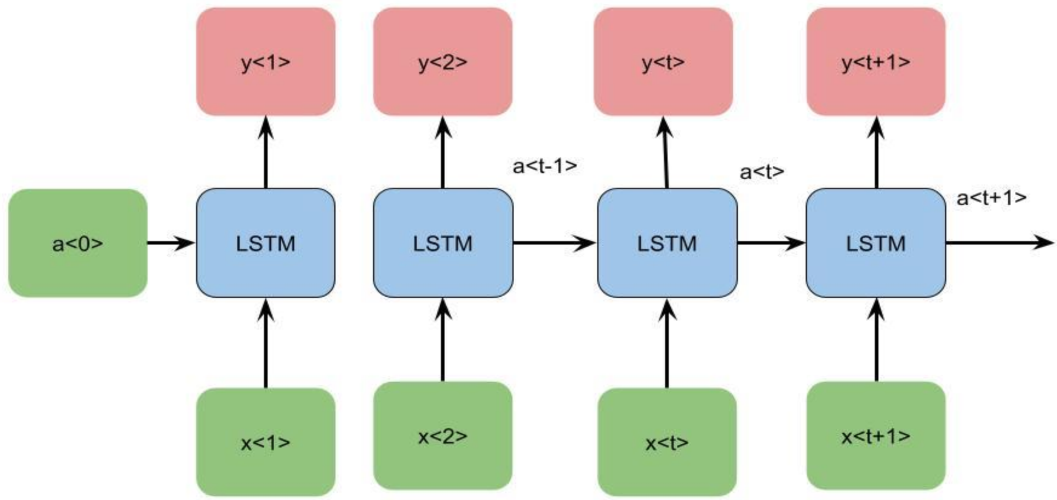


Figure-4 RNN for analysis of single-dimensional scans.

For the enhanced extraction of feature sets, RNN method combines a number of LSTMs. This LSTM functions by combining the variance and tangent maximisation functions to estimate various feature sets. Using scanned data as input, this model evaluates an initialization vector. through formula (6),

$$i = var\left(x_{in} * U^{i} + h_{t-1} * W^{i}\right) \quad (6)$$

In equation (7), var stands for variance levels, h for the activation matrix, xin for input scan values, and U and W for the LSTM process's activation hyperparameters. Equations (6) and (7) are used to assess initialization features, which are cascaded with functional features as well as output features.

$$f = var\left(x_{in} * U^{f} + h_{t-1} * W^{f}\right)$$

$$o = var\left(x_{in} * U^{o} + h_{t-1} * W^{o}\right)$$

$$C_t' = \tanh\left(x_{in} * U^{g} + h_{t-1} * W^{g}\right) \quad (7)$$

Using equation (8), the LSTM's final output features are extracted.

$$, T_{out} = var\left(f_t * x_{in}(t-1) + i * C_t'\right) \quad (8)$$

The crow conceals the location of its meal in its memory, which is the best possible candidate answer to the optimisation problem. Crows use algorithms to optimise their flight paths in orbit, where they constantly monitor aircraft in search of a better food supply. Assume that itermax is maximum number of iterations and N is size of population. x i,iter d in the i-th iteration represented the location of crow i in the d-dimensional search space, where =1,2,...,N and =1,2,...,itermax. MI,iter pointed out crow's hiding spot, which was by far its best location to yet. We might split the problem into two scenarios if we assumed that crow I followed crow J in each iteration to determine crow J's hidden position: 1) Crow j is tracking it without realising it, so Crow i can follow Crow j to retrieve concealed food; 2) Crow j realised that Crow i was tracking it, so Crow j will fly to a random spot in space to confuse Crow i and stop food from being stolen. The following was the expression of the Case 1 and Case 2 mathematical models by eqn (9)

$$c_{out} = \mathrm{SoftMax}\left(\sum_{i=1}^{N_f} f_i * w_i + b\right) \quad (9)$$

where APj,iter indicated the awareness probability of crow j at iteration. Crow's capacity to explore locally or globally will depend on the value of fl (fl1). Throughout the procedure, the ratio of the two distinct scenarios for the crow will be determined by the value of AP. Each crow's memory was refreshed concurrently using the subsequent formula (10)

$$\mathrm{Dim} = \mathrm{step} \times \frac{u \times \sigma}{|v|^{1/\beta}} \quad (10)$$

The crow search optimization flow chart is represented in Figure. 5

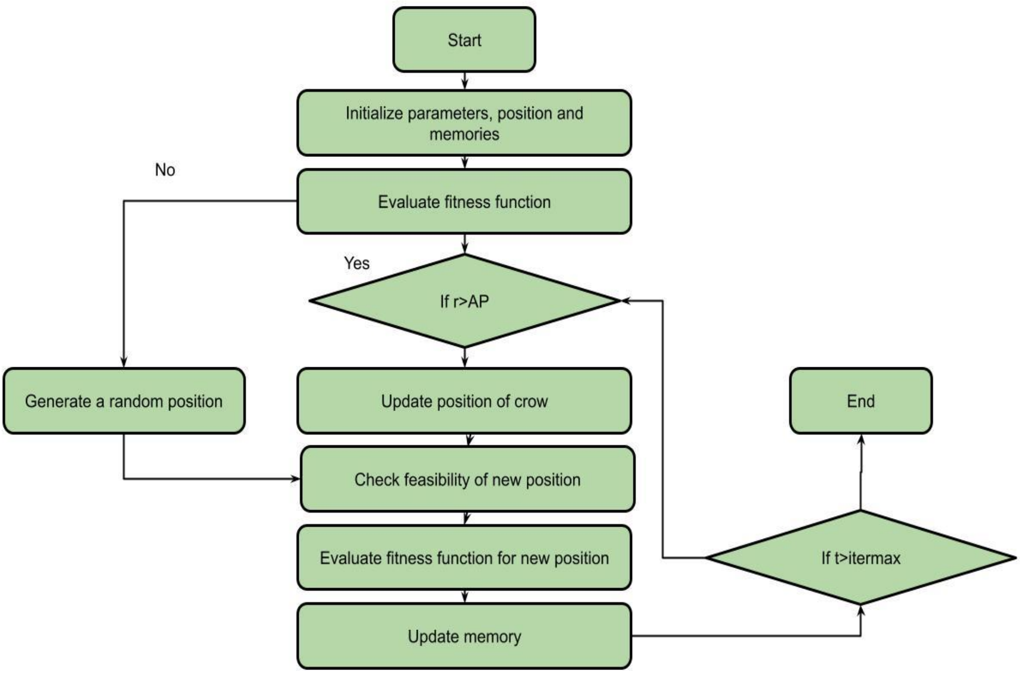


Figure-5 Flowchart for crow search optimization

The algorithm's rate of convergence will be slowed down by this kind of blind operation. We introduced the spiral search method as a solution to this issue. Crows have more opportunities to utilise their optimal location according to the spiral search mechanism, which improves their capacity to identify global solutions during the optimisation process. Distance between previous position as well as crow's best position was used to calculate trajectory of spiral structure, which served as a search space.

Given their characteristics, huge trevallies can travel great distances in search of their daily food, as was previously indicated. Therefore, eqn(11) is used in this stage to imitate the gigantic trevally's foraging movement patterns.

$$X(t+1) = \text{best p} \times R + (\text{Maximum} - \text{minimum} \times R + \text{Minimum}) \times \text{Levy(Dim)} \quad (11)$$

where BestP denotes current search space selected by gigantic trevallies based on best position found during their last search. Levy(Dim) refers to the Levy flight, a unique category of non-Gaussian stochastic processes, where the Levy distribution determines the step sizes. The algorithm's capacity to conduct a worldwide search is facilitated by sporadic huge steps. It is significant to note that a number of studies have demonstrated that a diverse range of species, including marine predators, exhibit Levy flying behaviour. Levy(Dim) should be computed with eqn(12):

$$\text{Levy (Dim)} = \text{step} \times \frac{u \times \sigma}{|v|^{1/\beta}} \quad (12)$$

To determine σ, use formula (13):

$$\sigma = \left(\frac{\Gamma(1+\beta) \times \text{sine}\left(\frac{\pi\beta}{2}\right)}{\Gamma\left(\frac{1+\beta}{2}\right) \times \beta \times 2^{\left(\frac{\beta-1}{2}\right)}} \right) \quad (13)$$

Giant trevallies locate and choose the optimal spot to look for prey in the designated search region based on the quantity of food (seabirds) in that area. Mathematically, this behaviour is simulated by equation (14).

$$X(t+1) = \text{best p} \times R + (\text{Maximum} - \text{minimum}) \quad (14)$$

where A is a position-change-controlling parameter with a range of 0.3 to 0.4, and X(t + 1) is the enormous trevally position vector in the subsequent iteration t. At time t (current iteration), the location of the enormous trevally i is Xi(t). In the meantime, Mean_Info, or the mean, may be computed using eqn(15) and shows that these huge trevallies have consumed all of the information that was accessible from the earlier spots.

$$\text{Mean info} = \sin(\theta_1^{\circ}) \times D \quad (15)$$

Using ten solutions (search agents) and five iterations, the Sphere function has been used to assess the efficacy of the picking area step, or (7). It is possible to estimate angle of incidence, angle of refraction given the knowledge of the incidence angle. The following illustrates Snell's law in eqn(16).

$$\eta_1 \sin\theta_1 = \eta_2 \sin\theta_2 \quad (16)$$

where θ1 and θ2 stand for angle of incidence and angle of refraction, and η1 = 1.00029 and η2 = 1.33 for the absolute refractive index of water and air. A random number in interval [0, 360] is denoted by θ2. From eqn(17)can be used to determine θ1:

$$\sin\theta_1 = \frac{\eta_2}{\eta_2} \sin\theta_2 \quad (17)$$

Visual distortion V can then be computed by applying eqn(18):

$$V = \sin(\theta_1^{\circ}) \times D \quad (18)$$

where D is distance between prey and attacker and may be computed utilizing equation (19): where sin is sine of a variable in degrees.

$$\mathrm{D} = |\text{best p} - \mathrm{Xi}(t)| \quad (19)$$

where BestP, which indicates the prey's location, is the best-obtained solution to date. Next, using eqn(20) a mathematical simulation is used to mimic the behaviour of gigantic trevally as they are chasing and jumping out of the water.

$$\mathrm{X}(t+1) = \mathrm{L} + \mathrm{V} + \mathrm{H} \quad (20)$$

where L, launch speed used to mimic following bird, determined utilizingeqn(21): where X (t + 1) is solution of subsequent iteration of t, which is given by attacking step.

$$\mathrm{L} = \mathrm{Xi}(t) \times \sin(\theta_1^{\circ}) \times \mathrm{F\_objXi}(t) \quad (21)$$

where fitness value of X at iteration t is denoted by F_obj(Xi(t)). The final term H in equation (22) can be computed using eqn(21) and represents the leaping slope function that allows method to adaptively execute a suitable transition from exploration phase to exploitation stage:

$$\mathrm{H} = \mathrm{R} \times \left(2 - t \times \frac{2}{\mathrm{T}}\right) \quad (22)$$

where R is a random number and here denotes various motion senses of huge trevally during exploitation stage, t and T stand for current iteration as well as maximum number of iterations. It is important to note that during iterations, H has a declining trend from 2 to 0, and that during the exploitation step, method attempts to exploit vicinity of solutions.

4. Results and discussion:

To simulate the topology, we use two systems, one with an Intel i5 3 GHz CPU and 8 GB of 1600 MHz DDR3 RAM, other with an Intel i7 2.7 GHz processor and 16 GB of 1600 MHz DDR3 RAM.

Dataset description: Twelve representative healthcare datasets are included in the current implementation; three are carried over from previous DIR version and are the Healthcare Cost and Utilisation Project (HCUP), Truven Health MarketScan (MarketScan), Medical Information Mart for Intensive Care (MIMIC); the remaining nine are chosen from working group notes that were reviewed by subject matter experts at UNC Charlotte Health Informatics and Outcomes Research Academy.National Longitudinal Study of Adolescent to Adult Health (Add Health), Premier Healthcare Database (Premier), Clinformatics Data Mart (Clinformatics), Clinical Practice Research Datalink (CPRD), Health Improvement Network (THIN), National Health and Nutrition Examination Survey (NHANES), and Humedica NorthStar (Humedica) are the nine extended datasets.

In the trials conducted using these settings, one IoT device was deployed every minute, transmitting ECG data packets, until a total of eight IoT devices were reached. That instance, one IoT device is operational in the first minute, and eight IoT devices are transmitting ECG data packets at the start of the eighth and final minute. After being turned on, every IoT device begins transmitting at 14 pps for configuration 1 and 128 pps for configuration 2. An Internet of Things device that is switched on stays turned on until experiment is over. Initial IoT device in the heterogeneous scenario (configuration 3) begins transmitting at a packet rate of 128 pps, while subsequent IoT devices alternately transmit at packet rates of 14 and 128 pps.

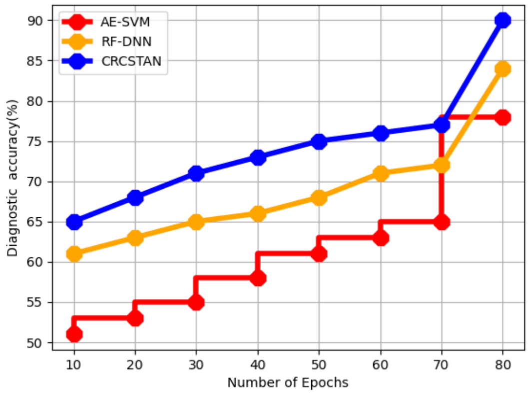
In order to prevent data congestion, t must satisfy condition t 6 (n/m), which needs to be taken into account from two angles that have an impact on t1 and t3: (1) The volume of data generated over same time period will rise in tandem with a progressive growth in the number of users. The sending delay t1 is directly impacted by whether the data processing point's network bandwidth allows for the transmission of such a big volume of data in a short amount of time. (2) The data processing point's performance is constrained. The subsequent portion of data is queued for processing, lengthening time t2, when the concurrently processed data reaches the data processing point's performance limit.

There are performance and network I/O bottlenecks in cloud as each thread's request volume progressively rises. Consequently, latency gradually increases. When the number of concurrent requests that each cloud platform thread must handle is less than 7, there is a delay of approximately 393 milliseconds. When the thread is processing seven or eight simultaneous requests, the delay approaches at least 384.7 ms. Requests must be queued once they reach the upper limit of what a single thread can handle. As a result, there will be significant data congestion, delays will improve until capacity to do real-time diagnostics is lost. As opposed to this, edge devices are dispersed throughout each user's house and can take the form of an intelligent electronic device, a gateway, a router, a smart gadget specifically designed for smart healthcare, or even the user's smartphone. They are all equipped with some sort of data computing capability.As number of users gradually improves, issues akin to data congestion in cloud platform won't arise, latency will consistently remain consistent at roughly 320 ms. Since there is typically at least one user at home, handling their healthcare data is made stable and effective.

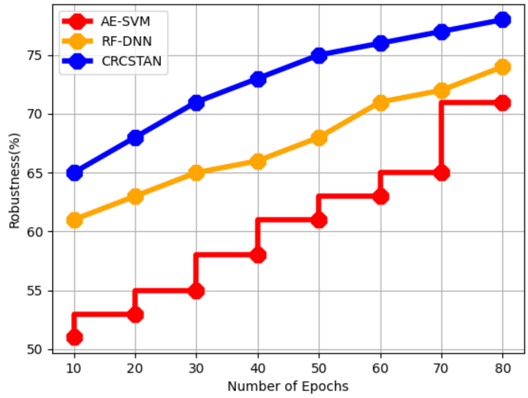
Table-1 Comparative for various smart healthcare dataset

Datasets	Techniques	Diagnostic accuracy	Robustness	Running memory	Power consumption	RMSE
HCUP	AE-SVM	78	71	48	55	55
	RF-DNN	84	74	50	59	53
	CRCSTAN	90	79	45	50	49
MIMIC	AE-SVM	76	72	54	57	65
	RF-DNN	86	75	60	53	61
	CRCSTAN	93	86	49	49	58
NHANES	AE-SVM	77	73	78	61	67
	RF-DNN	85	76	80	65	64
	CRCSTAN	96	89	70	58	57

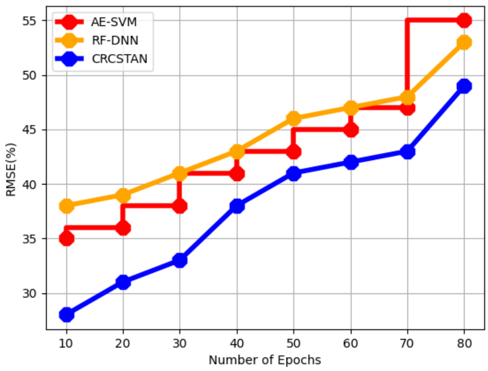
Table-1 shows comparative for various security datasets. here the dataset analysed are HCUP, MIMIC, NHANES. the parameter analysed are Diagnostic accuracy, robustness, running memory), Power consumption, RMSE.prognosis time is defined as amount of time ECG diagnosis model needs to process data before producing the final prognosis. This indicates that a quicker diagnosis speed can reduce the system's overall delay. Therefore, we ought to guarantee a reduced forecast time and get a basic accuracy that surpasses that of cardiologists. The 2017CinC optimum model's forecast time is approximately 100 ms longer than ours due to the intricate computations involved. A delay of this length is undoubtedly disastrous for field of smart healthcare. Because of this, our method has a distinct advantage in terms of the amount of memory used by runtime, inferred time, extent of storage space it occupies. This allows the model to be deployed with ease on edge devices with limited resources. It is compatible with our planned new cloud-edge-integrated home smart healthcare architecture.



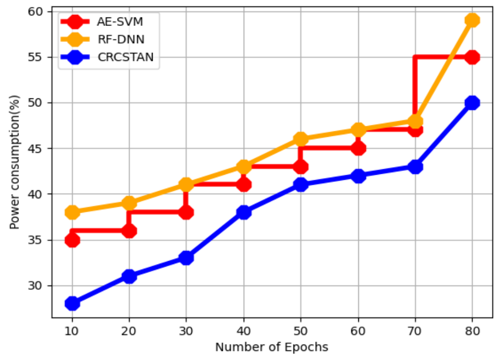
(a) diagnostic accuracy



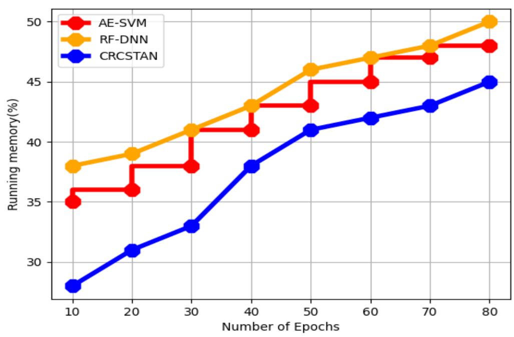
(b) robustness



(c) RMSE



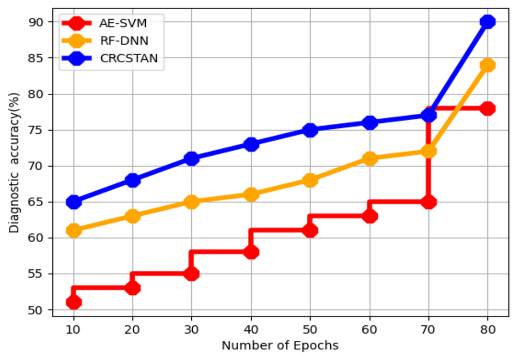
(d) power consumption



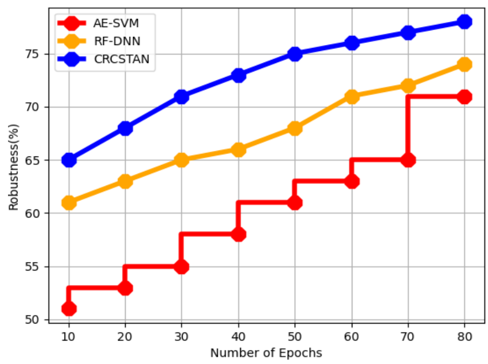
(e) running memory

Figure-6 Comparative for HCUP dataset(a) diagnostic accuracy, (b) robustness, (c) RMSE, (d) power consumption, (e) running memory

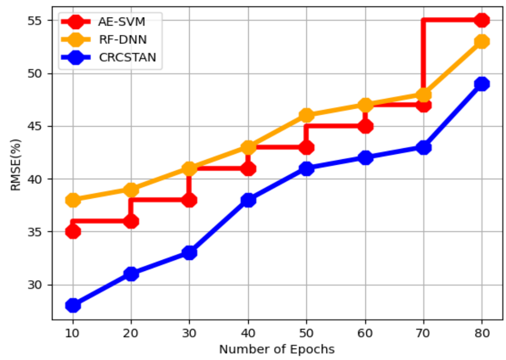
Figures 6(a)-(e) above compare suggested and current methods for the HCUP dataset. The proposed model attained diagnostic accuracy of 90%, robustness of 79%, RMSE 49%, power consumption of 50%, running memory 45%. While the existing AE-SVM achieved diagnostic accuracy of 78%, robustness of 71%, RMSE 55%, POWER CONSUMPTION of 55%, running memory 48%, RF-DNN achieved diagnostic accuracy 84%, robustness 74%, RMSE of 53%, POWER CONSUMPTION 59%, running memory 53%.



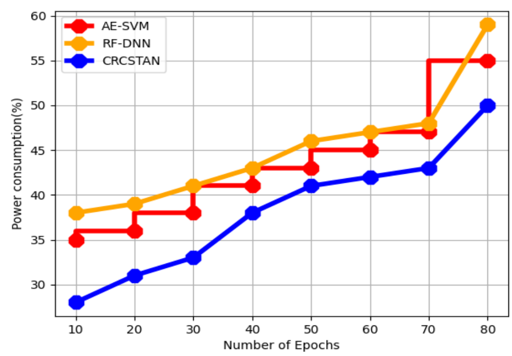
(a) diagnostic accuracy



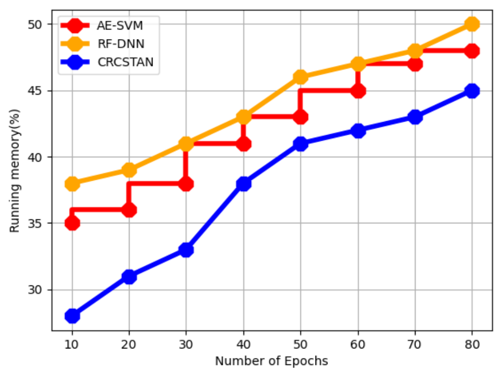
(b) robustness



(c) RMSE



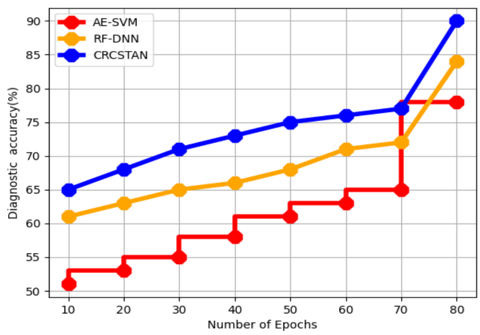
(d) power consumption



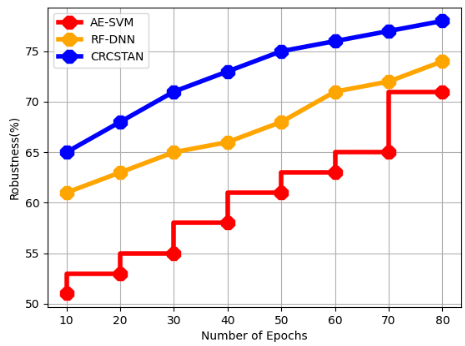
(e) running memory

Figure-7 Comparative forMIMIC dataset (a) diagnostic accuracy, (b) robustness, (c) RMSE, (d) power consumption, (e) running memory

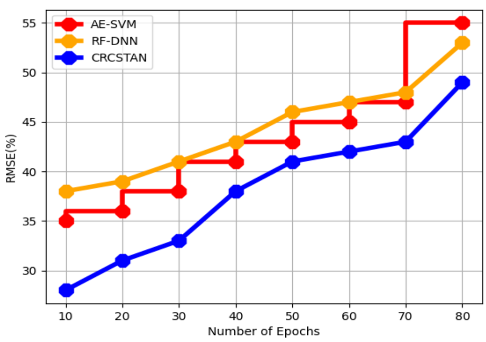
The MIMIC dataset-based comparison analysis of suggested and current approaches is displayed in figures 7(a) -(e) above. In this case, the suggested method achieved 93% diagnostic accuracy, 86% robustness, 58% RMSE, 49% POWER CONSUMPTION, 49% running memory; the current AE-SVM achieved 76% diagnostic accuracy, 72% robustness, 65% RMSE, 57% POWER CONSUMPTION, 54% running memory; RF-DNN achieved 86% diagnostic accuracy, 75% robustness, 61% RMSE, 53% POWER CONSUMPTION, 60% running memory.



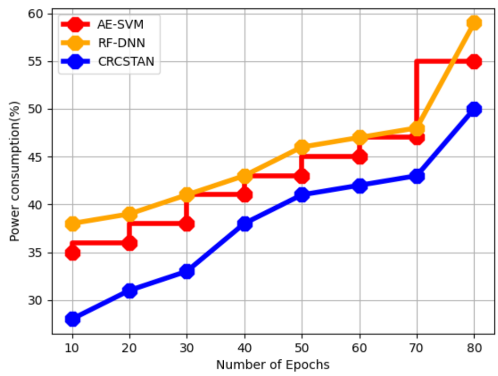
(a) diagnostic accuracy



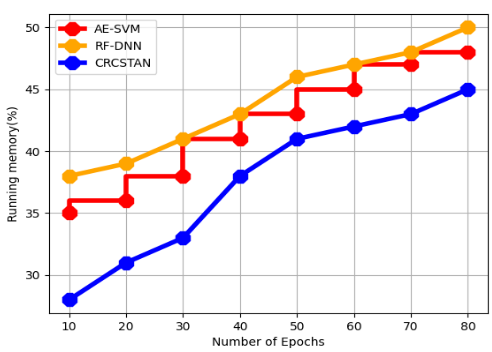
(b) robustness



(c) RMSE



(d) power consumption



(e) running memory

Figure-8 Comparative forNHANES dataset (a) diagnostic accuracy, (b) robustness, (c)

RMSE, (d) power consumption, (e) running memory

For the NHANES dataset, the comparison analysis between suggested and current methodologies is displayed in Figure 8(a)–(e) above. The suggested method achieved 96% diagnostic accuracy, 89% robustness, 57% RMSE, 58% POWER CONSUMPTION, 70% running memory; the current AE-SVM method achieved 77% diagnostic accuracy, 73% robustness, 67% RMSE, 61% POWER CONSUMPTION, 78% running memory; RF-DNN method achieved 85% diagnostic accuracy, 76% robustness, 64% RMSE, 65% POWER CONSUMPTION, 80% running memory.

Results of experiment show that suggested technique can reduce amount of data transferred to the cloud by up to 80% without sacrificing much accuracy. Even when reduction is performed on data that has been preprocessed using sliding windows, the accuracy is increased by the sliding window technique. While window sizes 25 lead to decreased accuracy for all circumstances (all sensors, location-based, and similarity-based), window sizes 50 and 100 yield similar accuracy. When the sliding panes are big enough, even a 90% reduction causes very little accuracy loss. Nevertheless, data aggregation on the edge is a drawback of the sliding window scenarios. Since sensor readings from distinct windows will be sent to the cloud for each time step in the sliding step one, network traffic will remain high. In order to prevent readings from belonging to distinct windows and to minimise network traffic, the sliding step must be equal to or larger than the window size; nonetheless, the application scenario must account for a window length delay. Reproduced data classification performed marginally better than reduced data classification; the difference grew for narrower windows.

5. Conclusion:

Using edge cloud computing and deep learning models for storage management, this research proposes a revolutionary approach to smart healthcare data analysis. This case involves the use of an IoT multi-sensor model to monitor healthcare data, which is then sent to an edge cloud computing model for storage. Continuous, replication-free indexing and time-limited information retrieval from storage are the two methods used to accomplish concurrency. Using deep learning, all storage instances' limitations are categorised for data augmentation and updating. A convolutional recurrent crow search Trevally adversarial network is used in this instance to handle data concurrency.We describe a cloud- and edge-based hybrid smart medical architecture. This architecture can safeguard data privacy in addition to cutting down on transmission overhead and diagnosis delay. Additionally, we create a deep learning model for ECG inference that works well and can be implemented on edge smart devices. A fair balance between diagnosis accuracy and resource cost is achieved by this diagnosis model. The experimental findings demonstrate that data enhancement broadens the diagnosis category in addition to improving the diagnosis model's overall accuracy. The following factors can be used to conduct additional research. More research is still needed to determine how medical morphological aspects of the ECG are represented in data. A basic, single ECG waveform can be used to diagnose certain cardiac disorders, but waveforms of many heart ailments frequently exhibit complex, varied morphological features. To create better diagnostic models, researchers must have a greater comprehension of these data.

References

1. Sharif, Z., Jung, L. T., Ayaz, M., Yahya, M., & Pitafi, S. (2023). Priority-based task scheduling and resource allocation in edge computing for health monitoring system. Journal of King Saud University-Computer and Information Sciences, 35(2), 544-559.
2. Hennebelle, A., Materwala, H., & Ismail, L. (2023). HealthEdge: a machine learning-based smart healthcare framework for prediction of type 2 diabetes in an integrated IoT, edge, and cloud computing system. Procedia Computer Science, 220, 331-338.
3. Singh, A., & Chatterjee, K. (2023). Edge computing based secure health monitoring framework for electronic healthcare system. Cluster Computing, 26(2), 1205-1220.
4. Meng, L., & Li, D. (2023). Novel Edge Computing-Based Privacy-Preserving Approach for Smart Healthcare Systems in the Internet of Medical Things. Journal of Grid Computing, 21(4), 66.
5. Khanh, Q. V., Hoai, N. V., Van, A. D., & Minh, Q. N. (2023). An integrating computing framework based on edge-fog-cloud for internet of healthcare things applications. Internet of Things, 23, 100907.
6. Jeyakumar, V., Abirami, K. R., Saraswathi, S., Kumaran, R. S., &Marthi, G. (2023). Secure medical image storage and retrieval for Internet of Medical Imaging Things using blockchain-enabled edge computing. In Intelligent Edge Computing for Cyber Physical Applications (pp. 85-110). Academic Press.
7. Sharma, D. K., Chakravarthi, D. S., Shaikh, A. A., Ahmed, A. A. A., Jaiswal, S., & Naved, M. (2023). The aspect of vast data management problem in healthcare sector and implementation of cloud computing technique. Materials Today: Proceedings, 80, 3805-3810.
8. Gao, H., Huang, H., Xue, L., Xiao, F., & Li, Q. (2023). Blockchain-enabled fine-grained searchable encryption with cloud-edge computing for electronic health records sharing. IEEE Internet of Things Journal.
9. Mahajan, H. B., & Junnarkar, A. A. (2023). Smart healthcare system using integrated and lightweight ECC with private blockchain for multimedia medical data processing. Multimedia Tools and Applications, 82(28), 44335-44358.
10. Rajagopal, S. M., Supriya, M., &Buyya, R. (2023). FedSDM: Federated learning based smart decision making module for ECG data in IoT integrated Edge-Fog-Cloud computing environments. Internet of Things, 100784.
11. Yan, S., He, L., Seo, J., & Lin, M. (2020). Concurrent healthcare data processing and storage framework using deep-learning in distributed cloud computing environment. IEEE Transactions on Industrial Informatics, 17(4), 2794-2801.
12. Sujith, A. V. L. N., Sajja, G. S., Mahalakshmi, V., Nuhmani, S., &Prasanalakshmi, B. (2022). Systematic review of smart health monitoring using deep learning and Artificial intelligence. Neuroscience Informatics, 2(3), 100028.
13. Bi, H., Liu, J., & Kato, N. (2021). Deep learning-based privacy preservation and data analytics for IoT enabled healthcare. IEEE Transactions on Industrial Informatics, 18(7), 4798-4807.
14. Xie, F., Yuan, H., Ning, Y., Ong, M. E. H., Feng, M., Hsu, W., ... & Liu, N. (2022). Deep learning for temporal data representation in electronic health records: A systematic review of challenges and methodologies. Journal of biomedical informatics, 126, 103980.
15. Praveen, S. P., Murali Krishna, T. B., Anuradha, C. H., Mandalapu, S. R., Sarala, P., & Sindhura, S. (2022). A robust framework for handling health care information based on machine learning and big data engineering techniques. International Journal of Healthcare Management, 1-18.
16. Amin, R., Al Ghamdi, M. A., Almotiri, S. H., &Alruily, M. (2021). Healthcare techniques through deep learning: issues, challenges and opportunities. IEEE Access, 9, 98523-98541.
17. Sammeta, N., & Parthiban, L. (2022). Hyperledger blockchain enabled secure medical record management with deep learning-based diagnosis model. Complex & Intelligent Systems, 8(1),

625-640.

18. Moqurrab, S. A., Tariq, N., Anjum, A., Asheralieva, A., Malik, S. U., Malik, H., ... & Gill, S. S. (2022). A deep learning-based privacy-preserving model for smart healthcare in Internet of medical things using fog computing. Wireless Personal Communications, 126(3), 2379-2401.
19. Qureshi, K. N., Din, S., Jeon, G., &Piccialli, F. (2020). An accurate and dynamic predictive model for a smart M-Health system using machine learning. Information Sciences, 538, 486-502.
20. Lie, W., Jiang, B., & Zhao, W. (2020). Obstetric imaging diagnostic platform based on cloud computing technology under the background of smart medical big data and deep learning. IEEE Access, 8, 78265-78278.
21. Kumar, R., Kumar, P., Tripathi, R., Gupta, G. P., Islam, A. N., &Shorfuzzaman, M. (2022). Permissioned blockchain and deep learning for secure and efficient data sharing in industrial healthcare systems. IEEE Transactions on Industrial Informatics, 18(11), 8065-8073.
22. Ali, F., El-Sappagh, S., Islam, S. R., Kwak, D., Ali, A., Imran, M., & Kwak, K. S. (2020). A smart healthcare monitoring system for heart disease prediction based on ensemble deep learning and feature fusion. Information Fusion, 63, 208-222.
23. Nazir, S., Khan, S., Khan, H. U., Ali, S., Garcia-Magarino, I., Atan, R. B., & Nawaz, M. (2020). A comprehensive analysis of healthcare big data management, analytics and scientific programming. IEEE Access, 8, 95714-95733.