International Journal of Computing and Digital Systems
2025, VOL. 17, NO.1, 1-15

http://dx.doi.org/10.12785/ijcds/1571033989

Bl-Boost:A Blockchain-based *XG*-Boost EHR Scheme In Healthcare *5.0* Ecosystems

Varun Deshmukh[1], Sunil Pathak[1] and Pronaya Bhattacharya[2]

[1]*Department of Computer Science and Engineering, Amity School of Engineering and Technology, Amity University, Rajasthan, Jaipur, India*

[2]*Department of Computer Science and Engineering, Amity School of Engineering and Technology, and Research and Innovation Cell, Amity University, Kolkata, West Bengal, India*

Received 29 May 2024, Revised 28 November 2024, Accepted 5 December 2024

Abstract: Healthcare 5.0 focuses on a personalized patient-centric approach, and combines advanced technologies like artificial intelligence (AI), blockchain, Internet-of-Things (IoT), and Big data to form preventive, proactive, and emotive healthcare. To assure privacy of electronic health records (EHRs) in Healthcare 5.0, blockchain has emerged as a disruptive technology owing to its properties of assured immutability, chronology, and transparent nature. Recent research has integrated blockchain technology with deep learning (DL) models to enhance the predictive capabilities for future disease occurrences. Nonetheless, DL models often necessitate a substantial volume of labeled data, a resource that may not be readily available in all scenarios.Thus, boosting mechanisms can overcome this limitation by leveraging small labelled datasets and improve the model generalization capability. Motivated by this, we propose a scheme, **Bl**ockchain based e**x**treme **g**radient (XG) **boost**ing scheme, where highlighted letters form the scheme acronym *Bl-Boost*. The scheme uses XG with long short term memory (LSTM), denoted as X-LSTM model for making accurate predictions on EHR data with the help of blockchain. We store the model predictions on a local interplanetary file systems (IPFS) server, and hash information is published in main blockchain. Via smart contracts (SCs), we aim for privacy-preserved access control on the data. The experimental validation is performed on the benchmark heart failure prediction dataset in terms of accuracy, loss, and precision matrix for LSTM and XG-Boost LSTM models. We validate the proposed scheme for validation accuracy and loss, EHR processing costs, IPFS scalability, mining latency, and resistance against collusion attacks. X-LSTM obtained an accuracy of 96.4% with 35 epochs, an 86% deployment time improvement over on-chain storage with IPFS, and a low latency of 50.23 milliseconds for 2500 transactions. The presented outcomes indicates that the scheme has strong potential for viability in real-world deployment scenarios.

Keywords: Blockchain, Deep Learning, Healthcare Analytics, Extreme Gradient Boosting, Long Short Term Memory

1 Introduction

Recently, the advent of Healthcare Internet-of-Things (HIoT) has led to the generation of enormous volumes of data, resulting in significant challenges in managing and processing data from various sources [1]. According to the International Data Corporation (IDC), global healthcare data is projected to reach 163 zettabytes by 2025 [2], driven by more devices and sensors. Electronic health records (EHRs) are crucial in modern healthcare, encompassing patients' medical history, treatments, medications, etc., but their volume challenges data processing and prediction [3]. Healthcare 4.0 systems focus on data integration, but varied formats and fragmentation lead to inaccurate analysis [4][5].

Healthcare 5.0 uses technologies like machine learning, big data analytics, and blockchain to extract insights from EHRs and provide personalized care [6]. It combines IoT protocols, fifth generation (5G) communication, and security solutions to create a patient-centric model. The use of blockchain assures a transparent and traceable EHR interoperability among healthcare systems, ensuring data integrity and minimizing errors and fraud.

Every transaction in EHR is recorded and traceable, reducing administrative costs. However, blockchain alone is not enough for Healthcare 5.0; effective artificial intelligence (AI) support is essential. Machine learning (ML) and deep learning (DL) techniques are widely used in EHR analysis. While ML and DL techniques have shown promising results in healthcare EHR analysis, there are still some limitations that need to be addressed. One of the significant limitations is the requirement of large amounts of high-quality data for training the models [7]. Another limitation is the difficulty in interpreting the results of the models [8]. Additionally, there are concerns regarding the potential for algorithmic bias and ethical issues in the use of these models [9].

Inspired by the preceding discussions, in this paper,

E-mail address: varundeshmukh007@gmail.com, sunilpath@gmail.com, pbhattacharya@kol.amity.edu

we propose a scheme, *Bl-Boost*, addresses several critical challenges in healthcare analytics, including the scarcity of labeled data for ML models, the need for real-time predictive capabilities, and the privacy concerns surrounding patient data. which integrates blockchain and XG-Boost to secure and manage EHRs. The scheme addresses the dual benefits of fast, reliable, and accurate predictive analysis. The combination of XG-Boost and LSTM is based on the complementary strengths of both models. While LSTM specializes in learning from sequential data, XG-Boost strengthens model performance by handling small and imbalanced datasets. Together, they create a robust and scalable model, suitable for healthcare applications with limited labeled data.

A. *Novelty*

The proposed scheme combines XG-Boost and LSTM allows operation with small labeled datasets. This integration enhances accuracy even with small and imbalanced datasets, with ensured security and transparent access control via blockchain. The integration presented as a stacked X-LSTM network (LSTM and XG-Boost) allows processing of the sequential healthcare data, capturing temporal dependencies such as trends in patient health records over time. These extracted features are then fed into the XG-Boost model, which enhances predictive accuracy by handling small, imbalanced datasets efficiently. Combined with the off-chain IPFS, it ensures decentralized data storage, reducing reliance on centralized servers, and secures the data through blockchain for immutability and integrity.

B. *Research objective and Contributions*

The article's research contributions encompass the following points.

- The proposed scheme is structured around a three-layered system model, which includes the following components: the data collection layer, the XG-Boost LSTM layer, and the blockchain and smart contract (SC) layer.
- Based on the layered model, the operation flow of the X-LSTM module that uses the stochastic and regularized gradient boosting features that ensures accuracy and execution speed of transactional analysis is formulated.
- Performance analysis for the X-LSTM model, and the blockchain metrics. The use of interplanetary file systems (IPFS) improves the mining rate, which is depicted in the results.

C. *Article Organizations*

The structure of the paper is as follows: Section 2 introduces key terminologies related to blockchain, healthcare analytics, gradient boosting, and reviews current state-of-the-art schemes. Section 3 details the system model and problem formulation. The proposed scheme is outlined in Section 4. Section 5 evaluates the performance of the *Bl-Boost* scheme. Finally, Section 6 concludes the paper.

2 Background and State-of-the-art

The section highlights the background of healthcare analytics, use of blockchain and IPFS in healthcare, XG boost mechanism, and related approaches. The details are presented as follows.

A. *Analytics and Blockchain in Healthcare 5.0*

Healthcare 5.0 shifts to a personalized, human-centric approach for proactive care. The healthcare industry faces challenges like aging populations, high costs, disease outbreaks, and chronic illnesses [10]. Healthcare analytics (HA) provides critical insights to address these issues and extend care reach.

Blockchain is a decentralized ledger where each block contains transactions linked by previous block hashes, forming a chronological trail of patient EHR histories. Any alteration invalidates subsequent block hashes, ensuring immutability, integrity, and reliability. It eliminates the need for centralized data collection, allowing multiple healthcare silos to manage data on a distributed network. Blockchain promotes transparency and access, with records accessible to authorized members in public, private, or hybrid setups. [11].

1) *Extreme Gradient Boosting*

Extreme Gradient Boosting (XGBoost) [12], an advanced ensemble gradient boosting method, has outperformed Friedman's gradient boosted trees and RF methods [13][14]. XGBoost's efficiency and fast training excel in both classification and regression tasks. Unlike RF, which uses randomized, diverse trees, gradient boosting combines weak learners into a strong one, sequentially building shallow trees where each corrects the previous ones. This reduces overfitting through a rule-based approach, while RF creates fewer, deeper trees. XGBoost advances traditional gradient boosting decision tree (GBDT) techniques by merging weak classifiers into a potent one using a classification and regression tree (CART) model. It sequentially adds trees, splitting features based on residuals. An unspent equation fits new residuals, aiming to accurately predict sample scores upon training completion.

Figure 1 depicts attributes pointing to analogous leaf nodes. This suggests that every tree will harbor its unique leaf node, with each corresponding to a specific score. To predict the sample's precise value, the cumulative scores from all the trees must be taken into consideration. Such nuanced execution represents a leap forward in machine learning, and underscores the agility and precision of XG-Boost, making it a favored approach for numerous applications in the realm of artificial intelligence.

B. *State-of-the-art*

This section furnishes an extensive overview of the pertinent methods, accompanied by a comparative assessment of their efficacy. TABLE I presents the state-of-the-art (SOTA)

TABLE I. Relative comparison of proposed scheme with state-of-the art approaches

Author(s)	Year	1	2	3	4	5	Advantages	Limitations
Kumar*et.al.* [15]	2018	N	Y	Y	N	-	This study presents a scalable three-tier IoT architecture for processing sensor data to identify crucial clinical parameters for heart disease detection. ROC (Receiver Operating Characteristic) analysis is used to pinpoint key clinical markers indicating impending cardiac conditions.	The architecture is bulky and not so much energy efficient if deployed for IoT systems.
Khan*et.al.* [16]	2018	N	Y	Y	N	92	The study reveals that Raman spectroscopy combined with ML can significantly aid in diagnosing and investigating infectious diseases.	The clinical practice to verify accuracy is still needed.
Amin *et.al.* [17]	2018	N	Y	Y	N	93	An automated technique for segmenting and discriminating brain tumors.	Adding more features can enhance the accuracy of the algorithm.
Zeng *et.al.* [18]	2019	N	Y	Y	N	-	The model in this paper combines features from unstructured and structured patient data for detecting breast cancer occurrences.	Clinicians often record ruled-out diagnoses or disputed symptoms, but this clinical narrative is not considered in the results.
Shao *et.al.* [19]	2019	N	Y	Y	N	90	CD codes alone are insufficient to detect dementia. The authors combined EHRs with patients' structured and unstructured records to determine the dementia risk score.	The study's patient population has more older males than females, potentially causing skewness and negatively impacting the results.
Bernardini *et.al.* [20]	2019	N	Y	Y	N	-	The model outperforms other SOTA competitors in terms of predicting performance and computation time, according to the results. Furthermore, the induced sparsity improves model inter-pretability by automatically managing high-dimensional data and the common imbalanced class distribution.	Nonlinear models with Gaussian functions are not considered here.
Allen *et.al.* [21]	2020	N	Y	Y	Y	89.09	This paper uses ensembled XG-Boost techniques, which outperformed other algorithms.	The sample size is small, and results may change with a larger population.
Le *et.al.* [22]	2020	N	Y	Y	Y	90.5	The algorithm created in this work could help with ARDS clinical trial recruitment as well as better ARDS prediction and early detection.	All results pertain to a single-center ICU setting. This study does not consider data from multiple centers or settings.
Budholiya *et.al.* [23]	2020	N	Y	Y	Y	91.8	The diagnostic approach in this paper improves decision-making quality during cardiac disease diagnosis.	The performance of the model tested for only one disease.
Chen *et.al.* [24]	2021	Y	Y	Y	N	-	The study introduced ML techniques for diabetes detection and secure data sharing with healthcare providers.	The patient data and doctors' data are stored in blockchain which make it bulky and processing delay occurs.
Shynu *et.al.* [25]	2021	Y	Y	Y	N	81	The article presents cost-effective, blockchain-based secure healthcare services, utilizing a feature selection-based adaptive neuro-fuzzy inference system to predict diabetes and cardiovascular diseases.	This paper does not consider the security and privacy of accessing patient medical data.
Kallimani *et.al.* [26]	2022	N	Y	Y	N	97.77	This article introduces an attention-based convolutional neural network (ACNN) combined with a long short-term memory (LSTM) model for heart disease detection, using novel feature selection techniques in a hybrid deep learning framework.	The ACNN and LSTM can give more accuracy if hyper-parameters are used effectively.
Neelakandan *et.al.* [27]	2022	Y	Y	Y	N	95.29	The article presents a model called Blockchain with DL-Enabled Secure Medical Data Transmission and Diagnosis (BDL-SMDTD) for disease diagnosis using medical images, ensuring secure data transmission via blockchain technology.	This is proposed methodology but clinical practice is missing is not yet done to check the accuracy.
Malibari *et.al.* [28]	2023	N	Y	Y	N	93.5 and 94	This article introduces the EO-LWAMCNet model, an optimized Lightweight Automatic Modulation Classification Network, for precise prediction of kidney and heart diseases in patients.	The execution time of EO-LWAMCNet model high compared to the existing models.
Alshraideh *et.al.* [29]	2024	N	Y	Y	N	94.3	This article employs a support vector machine (SVM) classifier integrated with particle swarm optimization (PSO) to conduct feature selection.	The study prioritizes the accuracy of the prediction model. However, additional metrics such as sensitivity, specificity, and the AUC-ROC could offer highly favored understanding of the model's performance.
Oladele *et.al.* [30]	2024	Y	Y	N	N	-	The author used hyper-fabric ledger blockchain for patient record management and maintaining transparency with the help of smart contracts and uses value pair is used to identify each record	The study privacy of the user and worked only population of 500.
Velmurugan *et.al.* [31]	2024	Y	Y	N	N	-	Blockchain-based hyper ledger fabric is used for the effective exchange of health records over the unsecured channel; suggested algorithms for data transformation are highly responsive.	Integrating data in blockchain from several sources creates security and legitimacy issues.
Kumar *et.al.* [32]	2024	Y	Y	N	Y	-	The framework's security is validated through formal and informal analysis, along with simulations via the Scyther tool. It outperforms existing solutions in terms of communication overhead, computation cost, and processing time.	The complexity of integrating the system with existing EHR infrastructures, and the computational resource demands for running the security validations and simulations.
Proposed	2024	Y	Y	Y	Y	96.4	A secured and scalable healthcare analytics by integrating XG-Boost and LSTM (X-LSTM) on labelled datasets.	Security validation is not considered as part of this study.

Parameters- 1. Blockchain 2. Health-care Analytics 3. Learning Models 4. Boosting Technique 5. Accuracy(%) Y- shows that the parameter is present, N- shows that the parameter is absent

approaches. The technical discourse is segregated based on the method employed, such as ML in healthcare analytics or blockchain in healthcare analytics.

1) ML in Healthcare Analytics

Recent progress in HIoT (Healthcare Internet of Things) integration has enabled remote monitoring and real-time

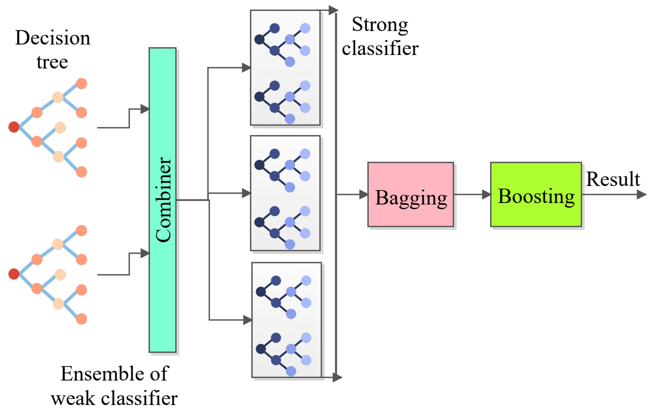


Figure 1. XG-Boost mechanism

tracking [33]. Managing the vast data from HIoT devices is challenging. AI integration helps diagnose, analyze, and detect diseases accurately, with algorithms predicting diseases swiftly in early stages [34].

AI has significantly contributed to disease diagnosis, analysis, and detection, resulting in more accurate disease classification. Kumar *et al.* [15] proposed a scalable architecture for processing sensor data in a three-tier IoT-based framework that prioritizes critical clinical parameters for heart disease detection. ROC analysis is used to identify the most important clinical markers that suggest an imminent cardiac condition. Khan *et al.* [16] presents the integration of Raman spectroscopy with ML, which can be highly beneficial in diagnosing and exploring infectious diseases. Amin *et al.* [17] proposed automated technique for segmenting and discriminating brain tumours. Authors in [18] proposed ML models for breast cancer detection based on unstructured and structured patient data. Authors in [19] used the International Classification of Diseases (ICD) codes on EHRs for dementia detection and computed the risk scores for the patients. In [21], authors proposed an XG-boost based technique to improve the accuracy for disease detection, and it performs better than conventional models.

2) Blockchain in Healthcare Analytics

Blockchain enables patient-centered healthcare through collaborative, transparent medical data management, ensuring patient privacy while allowing access for stakeholders. Burniske *et al.* highlighted its expanded use beyond cryptocurrencies [35]. Shynu *et al.* proposed a blockchain-based healthcare service for predicting diabetes and cardiovascular diseases within a fog computing framework [25]. It collects health data from fog nodes, securely stores it on the blockchain, clusters records using a rule-based algorithm, and forecasts diseases with a feature selection-based adaptive neuro-fuzzy inference system (FS-ANFIS). Neelakandan *et al.* presented a model using blockchain for secure medical data transmission and deep learning for diagnosis [27]. This model encrypts and stores images on the blockchain, employing histogram-based segmentation, feature extraction with Inception ResNetv2, and disease classification through a support vector machine (SVM), validated with benchmark medical images.

Many existing systems experience difficulties when scaling up to accommodate large datasets, resulting in increased latency and slower data retrieval. While these systems provide secure data storage, most do not incorporate predictive analytics to enhance decision-making, focusing instead on data management. Some systems have inefficient mechanisms for sharing data, often lacking automated access controls via smart contracts, which hampers secure and effective data exchange.

C. Strengths and Weakness of existing models

1) XG-boost: XG-Boost is a powerful gradient boosting algorithm well-suited for dealing with imbalanced data and small labeled datasets. It excels at capturing complex patterns in structured data, such as patient demographics and diagnostic information, which are often found in healthcare datasets. However, XG-Boost struggles with sequential or time-dependent data, which is essential in healthcare when analyzing changes in patient health over time.
2) LSTM: LSTM is ideal for processing time-series data, making it perfect for analyzing continuous patient information, like medical history and vital signs over a period of time. Its memory retention allows it to capture important temporal relationships in patient data. LSTM models often require large labeled datasets to perform optimally and can be prone to overfitting when dealing with smaller datasets.

Other models, such as Random Forests or standard neural networks, were not selected due to their lack of time-series processing capabilities or their inefficiency in handling imbalanced datasets. CNNs, while powerful for image data, are not as suitable for the kind of tabular and sequential data often found in healthcare.

3 System Model and Problem Formulation

This section outlines the system model and formulates the problem.

A. System Model

This section presents the system model of the proposed *Bl-Boost* scheme, integrating a blockchain-assisted solution for predictive analysis. Figure 2 shows the layered model with three layers: L_1 (data collection), L_2 (XG-Boost enabled LSTM), and L_3 (blockchain and SC). The details are as follows.

1) L_1: Data Collection Layer

- At this layer, we consider Healthcare Users (HU), including entities $E = E_p, E_d, E_{ia}, E_{lb}, E_{ahp}, E_a$: patients ($E_p$), doctors ($E_d$), insurance agents ($E_{ia}$), lab workers ($E_{lb}$), allied healthcare staff (E_{ahp}), and administrators (E_a). Patients' (E_p) EHRs (lab reports, prescriptions, insurance bills, claims) are secondary data (D_s). Primary data (D_p)

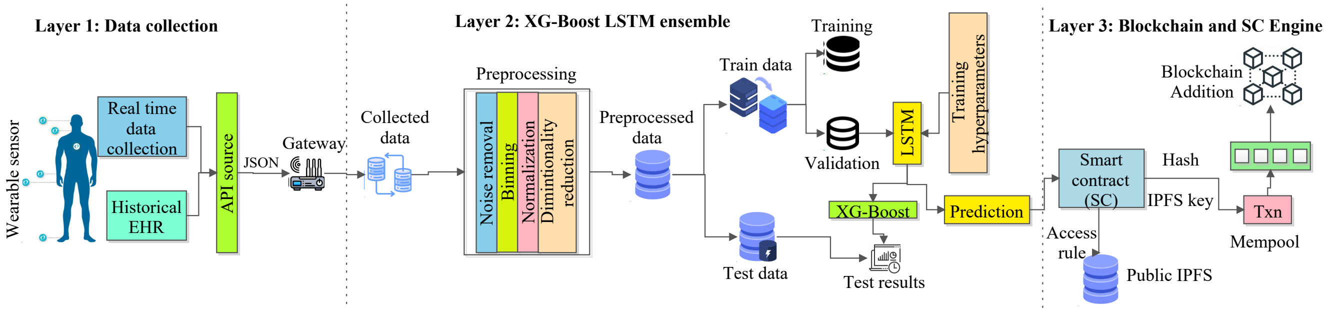


Figure 2. *Bl-Boost*: System Model

comes from sensors like blood glucose, electrochemical, and amperometric biosensors. Data is processed using sensor fusion algorithms for uniform readings [36]. Sensor data at L_1, combining D_p and D_s, is accessed via APIs in JSON format. Collected data is $D_c = D_1, D_2 \ldots, D_n$, mapped as $M: D_c \longrightarrow D_b$. D_c is sent to L_2 for preprocessing, cleaning, and reduction.

2) L_2: XG-Boost LSTM ensemble

- At L_2, the primary objective is to form predictions on the collected data. For the same, an ensemble of XG-Boost and LSTM (X-LSTM) is proposed. Initially, the data D_c undergoes the preprocessing stage, where it undergoes several transformations. At the first step of preprocessing, any outliers or unwanted noise from D_c are eliminated, which leads to a clean dataset D_{clean}. D_{clean} is then subjected to binning, where the continuous values are converted into discrete bins resulting in D_{binned}. To ensure uniformity in feature scales, this binned data is normalized, giving D_{norm}. Further, to enhance computational efficiency and possibly counteract overfitting, dimensionality reduction is applied to D_{norm}, producing the reduced data setD_{red}.

Post preprocessing, the processed data (D_{red}) is split into training (D_{red_train}) and test data (D_{red_test}). D_{red_train} is further split into training and validation data for the LSTM model, parameterized by hyperparameters θ (learning rate η, batch size $\mathbb{B}$, and number of epochs N). After LSTM training, the extracted features ($\mathcal{F}$) serve as input for the XG-Boost algorithm. This ensemble leverages LSTM's sequence processing and XG-Boost's predictive power, improving generalization and accuracy. Final predictions ($\mathcal{P}$) are derived from the LSTM and XG-Boost ensemble.

3) L_3: Blockchain and SC Engine

At L_3, the goal is to securely store prediction results ($\mathcal{P}$) using blockchain technology. Predictions are added to a decentralized blockchain database (B) for tamper-resistant storage and traceability. Access and interactions with this blockchain are governed by smart contracts (SCs).

For efficient retrieval and verification, prediction results are hashed, creating a unique identifier ($\mathcal{H}$), and stored in IPFS offline storage. Users access IPFS with a 32-byte content key (C_{key}). Through SCs, users retrieve prediction data from IPFS using C_{key} and its private identifier ($Pri(K)$). The C_{key} information is mapped to the IPFS record, with the key reference stored on the blockchain. Transactions are temporarily held in the Mempool ($\mathcal{M}$) before being confirmed and added to a block.

B. Problem Formulation

This subsection formalizes the objectives for the proposed *Bl-Boost* scheme, addressing challenges and constraints. Goals include enhanced accuracy, expedited predictive analysis, and minimized blockchain transaction sizes. The details can be presented as follows.

- Accuracy Enhancement in Ensemble Predictions: Given the ensemble of LSTM and XG-Boost, our first goal is to optimize the predictive accuracy. Let the prediction accuracy be denoted by A, which is a function of the features extracted by LSTM, $\mathcal{F}$, and the XG-Boost model parameters, θ. The objective can be expressed as follows.

$$P_1 : \max_{\theta} A(\mathcal{F}, \theta) \tag{1}$$

subject to constraint C_1 pertaining to the underlying data distribution, the capabilities of the LSTM, and the optimization landscape of the XG-Boost.

- Expedited Predictive Analysis: The computational efficiency is of paramount importance for real-time healthcare applications. Let $T(\mathcal{F}, \theta)$ represent the time taken by the X-LSTM ensemble to generate predictions. Our goal is to minimize T while maintaining a certain level of accuracy, A_{min}. Mathematically, it can be presented as follows.

$$P_2 : \min_{\theta} T(\mathcal{F}, \theta) \tag{2}$$

subject to constraint C_2 which specifies

$$A(\mathcal{F}, \theta) \geq A_{min} \tag{3}$$

- Minimization of Transaction Size in Blockchain: With the intent to create an efficient and scalable blockchain-assisted solution, we seek to minimize the transaction size. Denote the transaction size as S_{tx}, and the prediction result size as $S_{\mathcal{P}}$. By hashing the results and utilizing the IPFS storage, the goal is to minimize the effective transaction size added to the blockchain. It can be presented as follows.

$$P_3 : \min S_{tx}(\mathcal{H}, C_{key}, \mathcal{P}) \tag{4}$$

subject to constraint C_3, specified as follows.

$$S_{tx} \propto S_{\mathcal{P}} \tag{5}$$

This relation indicates that as the prediction result size grows, the transaction size should grow proportionally, but with mechanisms in place to keep it minimal.

Thus, the overall problem P_f can be treated as a minimization problem $min(-P_1, P_2, P_3)$ subject to the given constraints $\{C_1, C_2, C_3\}$.

C. The Multi Objective Optimization

Given our multi-objective function P_f, the Pareto Optimal solution set, denoted as $\mathcal{P}^*$ is defined as follows.

$$\mathcal{P}^* = \{x \in \mathcal{X} \mid \nexists x' \in \mathcal{X}\} \tag{6}$$

such that $f_i(x') \leq f_i(x) \forall i$ and $f_j(x') < f_j(x) \exists j$, where $f_i(x)$ is the i^{th} objective of P_f, and $\mathcal{X}$ is the feasible solution space defined by the constraints $C = \{C_1, C_2, C_3\}$. The above definition establishes that any solution $x^* \in \mathcal{P}^*$ is Pareto Optimal if no other feasible solution x' exists that can improve at least one objective without deteriorating any other objectives.

Now, to address the multi-objective optimization problem in the context of the X-LSTM model, we propose an optimization technique denoted as O_{XL}. This technique guides the model's parameters θ to achieve a balance among our objectives. Specifically, we incorporate the Pareto Optimal principle into the learning algorithm of the X-LSTM. Mathematically, the optimization problem can be expressed as follows.

$$O_{XL}(\theta) : \min_{\theta}(-P_1(\mathcal{F}, \theta), P_2(\mathcal{F}, \theta), P_3(\mathcal{F}, \theta)) \tag{7}$$

Proof: To demonstrate that our proposed solution O_{XL} effectively addresses the multi-objective optimization, three conditions are to be satisfied.

- Completeness: Given constraints C_1, C_2, C_3, our convex and bounded solution space $\mathcal{X}$ ensures a finite Pareto front from $\mathcal{P}^*$.
- Optimality: Each solution x from the Pareto front optimizes at least one objective without compromising others. By using O_{XL} in the X-LSTM model, the learning process converges to Pareto front solutions, ensuring multi-objective optimality.
- Efficiency: O_{XL}, tailored for the X-LSTM model, considers the structure of both LSTM and XG-Boost components, efficiently exploring $\mathcal{X}$ without unnecessary computations.
- Decomposition: O_{XL} breaks down the multi-objective problem into simpler subproblems, each targeting one objective while maintaining the others. This iterative approach generates Pareto-optimal solutions without exhaustively evaluating the entire solution space.
- Scalability: The decomposition approach allows O_{XL} to scale with data size and complexity, adapting dynamically to changing data distributions and conditions. If an objective becomes more critical due to external factors, optimization can refocus on that objective without restarting.

4 Bl-Boost: The Proposed Scheme

In this section, we delve into the proposed scheme. As indicated in previous section, we outline the ensemble of LSTM and XG-Boost, which present the optimal O_{XL} solution to the optimization problem.

A. The interaction flow

As presented in section 3-A, the proposed scheme forms a layered solution ($L_1 to L_3$), that presents a robust solution for healthcare analytics in Healthcare 5.0. Figure 3 presents the interaction flow of the proposed scheme. The raw data (processed EHR and real-time data) collected at L_1 in a heterogeneous manner is sent to algorithm 1 that provides the removal of outliers and noise. The step involves binning, and min-max normalization is applied. Next, we compute the dimensionality reduction, and the data is sent to the stacked X-LSTM model at L_2. Now, based on the presented optimization, we present the stacked X-LSTM model that applies the objective functions and sequentially processes the preprocessed data, capturing temporal dependencies and learning complex patterns over time. The network avoids overfitting or bias in its predictive outcomes. This is achieved by iteratively improving the model's parameters through multi-objective optimization, balancing key metrics such as prediction accuracy, computational efficiency, and blockchain transaction size.

At L_3, these predictions are securely integrated into the blockchain framework. The predictions are hashed and stored on the blockchain using a content-addressable IPFS, while SCs enforce privacy-preserved access control and data sharing policies. Smart contracts automate the verification of access rights, ensuring that only authorized users can retrieve or interact with the predictions. The interplay is synergistic; algorithm 1 ensures that L_2 receives data that is clean and computationally manageable. Algorithm 2 at L_2 optimizes predictions by interfacing with the Stacked X-LSTM network, and L_3 secures and stores these predictions in a decentralized and controlled environment among different healthcare stakeholders, governed by smart contracts. The details of the interaction flow are presented in the following subsections.

B. Data Preprocessing

The collected data D_c first undergoes outlier removal and noise reduction. We adopt the Interquartile Range (IQR) approach. Let Q_1, and Q_3 be the first and third quartiles of

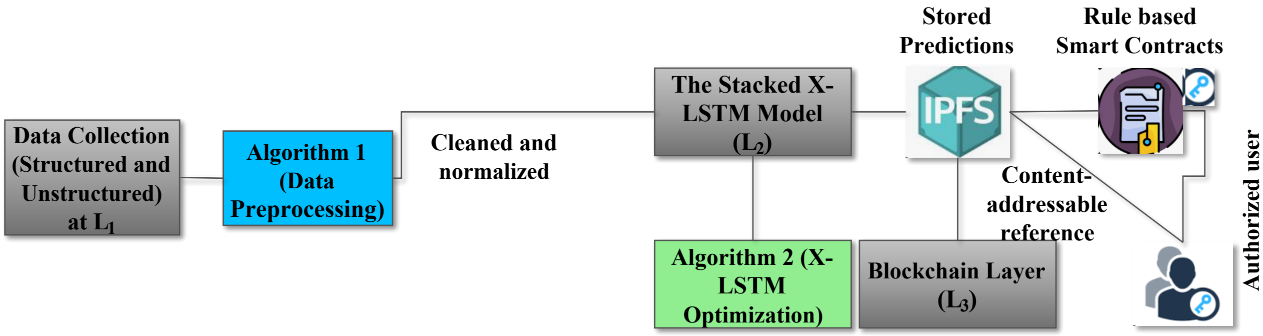


Figure 3. *Bl-Boost*: The interaction flow model

D_c. The IQR is then calculated as follows.

$$IQR = Q_3 - Q_1 \tag{8}$$

Any data point d from D_c that falls outside the range $[Q_1 - 1.5 \times IQR, Q_3 + 1.5 \times IQR]$ is considered an outlier and is thus removed. The resultant dataset post this filtration is D_{clean}.

After cleaning, data may still have fine-grained continuous attributes. Binning discretizes these values. Let the number of bins be B. The data range for each attribute in D_{clean} is divided into B equal-width intervals. The width is given as follows.

$$w = \frac{\max(D_{clean}) - \min(D_{clean})}{B} \tag{9}$$

Each interval represents a bin, and continuous values within an interval are replaced by a representative value, often the bin's mean or median. This results in D_{binned}.

To ensure uniform feature scales, Min-Max normalization is applied. For each feature $F \in D_{binned}$, normalization is performed as follows.

$$F_{norm} = \frac{F - \min(F)}{\max(F) - \min(F)} \tag{10}$$

Here, $\min(F)$, and $\max(F)$ are the minimum and maximum values of the feature $F \in D_{binned}$. The resulting dataset post-normalization is D_{norm}.

Next, we apply Principal Component Analysis (PCA), which finds orthogonal axes (principal components) that maximize data variance. If D_{norm} has m features, and we wish to reduce it to k dimensions, PCA finds k principal components such that $k < m$. The transformed data is then given as follows.

$$D_{red} = D_{norm} \times P \tag{11}$$

where P is the matrix with columns corresponding to the first k principal components of D_{norm}. The components in P are ordered by the amount of variance they capture from the original data. Typically, k is chosen such that a significant proportion (often 95% or more) of the total variance in the original data is retained. Mathematically, this can be represented as follows.

$$\sum_{i=1}^{k} \lambda_i \geq 0.95 \times \sum_{i=1}^{m} \lambda_i \tag{12}$$

Here, λ_i represents the eigenvalues of the covariance matrix of D_{norm}, sorted in descending order. The first k eigenvalues correspond to the variance explained by the first k principal components. The reduced dataset, D_{red} is of lower dimensionality, and preserves the majority of crucial information from the original dataset. This reduces potential overfitting, and ensures that the most significant patterns in the data are retained for predictive modeling.

Algorithm 1 details preprocessing with four functions: *RemoveOutliers* using the IQR method, *Binning* partitions D_{clean} into n equal-width intervals, transforming each into a discrete bin for easier computation, *Normalize* scales data to zero mean and unit variance, aiding scale-sensitive algorithms, and *ReduceDimensionality* employs PCA. Outlier removal, binning, and normalization operate at $O(n)$ per feature, while PCA's eigen decomposition of the covariance matrix is typically $O(d^3)$. Overall complexity is $O(d \times n + d^3)$, with n as the number of data points.

C. The stacked LSTM Network

In this subsection, we discuss the schematics of the stacked LSTM network. We consider that preprocessed data D_{red} is splitted into training and testing data, where the training data is fed to the stacked LSTM network. Figure 4 presents the details of the stacked LSTM network. For a single LSTM cell, the forget gate f_t in a LSTM cell decides the amount of the previous cell state to retain. The cell state C_t acts as the memory of the LSTM unit. It has the capability to store and retrieve information across extended sequences. Finally, the output gate o_t controls how much of the current cell state makes it to the hidden state.

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \tag{13}$$

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \tag{14}$$

$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C) \tag{15}$$

Algorithm 1 Preprocessing for D_c

Input: D_c: Collected data set, k: Number of principal components to retain, such that k 95% variance is retained.
Output: - D_{red}: Reduced data set after preprocessing.

Procedure Preprocess(D_c, k)
$D_{clean} \leftarrow RemoveOutliers(D_c)$
$D_{binned} \leftarrow Binning(D_{clean})$
$D_{norm} \leftarrow Normalize(D_{binned})$
$D_{red} \leftarrow ReduceDimensionality(D_{norm}, k)$
return D_{red}

Function $RemoveOutliers(D_c)$
for (each feature $f \in D$) **do**
  Compute $Q1$ and $Q3$
  $IQR \leftarrow Q3 - Q1$
  Remove data points where $f < Q1 - 1.5 \times IQR$ or $f > Q3 + 1.5 \times IQR$
end for
return D_{clean}

Function $Binning(D_{clean})$
for (each feature $f \in D_{clean}$) **do**
  Partition f into n equal-width intervals
  Convert each interval into a discrete value representing the bin
end for
return D_{binned}

Function $Normalize(D_{binned})$
for (each feature $f \in D_{binned}$) **do**
  $\mu_f \leftarrow$ mean of f
  $\sigma_f \leftarrow$ standard deviation of f
  $f_{norm} \leftarrow \frac{f-\mu_f}{\sigma_f}$
end for
return D_{norm}

Function $ReduceDimensionality(D_{norm}, k)$
Compute the covariance matrix Σ of D
Compute the eigenvalues λ and eigenvectors v of Σ
Sort λ in descending order and select the top k eigenvectors to form matrix P
$D_{red} \leftarrow D \times P$
return D_{red}

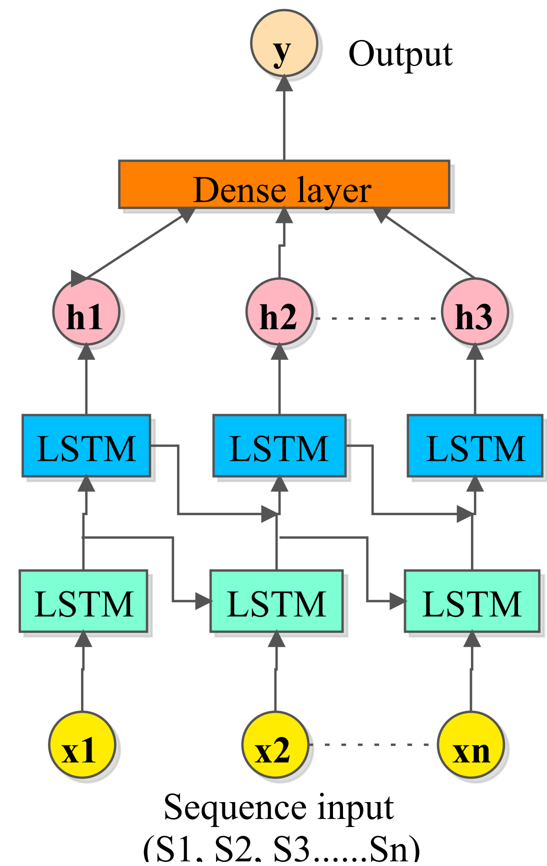


Figure 4. The stacked LSTM model

$$C_t = f_t \times C_{t-1} + i_t \times \tilde{C}_t \tag{16}$$
$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \tag{17}$$
$$h_t = o_t \times \tanh(C_t) \tag{18}$$

where $\tilde{C}_t$ denotes the new memory creation of the LSTM cell, C_t is the update cell state, h_t denotes the current hidden state, $[h_{t-1}, x_t]$ represents the concatenation of the previous hidden state and the current input, o_t is the output gate, σ denotes the sigmoid activation function, which squashes the output between 0 and 1. *tanh* is the hyperbolic tangent activation function, which outputs values between -1 and 1. $\{W_f, W_i, W_C, W_o\}$ are weight matrices for the forget gate, input gate, new memory, and output gate respectively, and $\{b_f, b_i, b_C, b_o\}$ are bias terms for the forget gate, input gate, new memory, and output gate respectively.

The input sequence $S_1, S_2, \ldots, S_n$ is divided into n input gates, where each gate i_t determines the stored information. LSTM units are stacked, with the output h_t from one unit becoming the input for the next. Assuming there are L LSTM layers, the operations for layer l are as follows.

$$h_t^{(l)} = \text{LSTM}(h_t^{(l-1)}, x_t) \tag{19}$$

where $h_t^{(0)}$ is the initial input to the LSTM network, x_t. After passing through all L LSTM layers, the final hidden state $h_t^{(L)}$ is fed into a dense layer to produce the final output y. The dense layer can be represented as follows.

$$y = \text{softmax}(W_d \cdot h_t^{(L)} + b_d) \tag{20}$$

where W_d is the weight matrix for the dense layer, b_d is the bias for the dense layer. The softmax function ensures that the output is a probability distribution over the target classes.

The complexity of an LSTM operation mainly depends on the size of the weight matrices. Given an input dimension d, and hidden state dimension h, the complexity of LSTM operations for each time step and each layer is $O(h \times d + h^2)$. Given T time steps and L layers, the total complexity becomes $O(T \times L \times (h \times d + h^2))$. The dense layer's complexity is $O(h \times c)$, where c is the number of output classes. Thus, the total complexity for the entire stacked LSTM network for all time steps is $O(T \times L \times (h \times d + h^2) + h \times c)$. The complexity analysis of the stacked LSTM network reveals its inherent computational demands, especially as the number of layers L and time steps T increase.

D. The X-LSTM network

In this subsection, we present the integration of the LSTM output y to be fed to the XG-Boost module. Given a sequence of data $S = \{s_1, s_2, \ldots, s_n\}$, the LSTM processes this sequence to produce a higher-level representation or embedding, represented as follows.

$$E = LSTM(S) \tag{21}$$

where $S$is the input sequence, and $E$is the embedding or output representation from the LSTM. The embedding E obtained from the LSTM serves as the input feature vector for the XG-Boost model, denoted as follows.

$$F_{XGB} = XGBoost(E) \tag{22}$$

where F_{XGB} is the prediction or output from the XG-Boost model. For the XG-Boost model, we set an initial prediction value for every observation, denoted as follows.

$$\hat{y}_i^{(0)} = \frac{1}{2}\log\left(\frac{\sum_{i=1}^{n} w_i y_i}{\sum_{i=1}^{n} w_i(1-y_i)}\right) \tag{23}$$

where $\hat{y}_i^{(0)}$ is the initial prediction for the i^{th} observation, w_i is the weight for the i^{th} observation, and y_i is the actual value for the i^{th} observation. In XG-Boost, we consider M trees, and we run iteratively $m = 1 to M$ and compute the Gradient and Hessian for the loss function. Thus, for each observation i, we have

$$g_i = \frac{\partial L(y_i, \hat{y}_i^{(m-1)})}{\partial \hat{y}_i^{(m-1)}} \tag{24}$$

$$h_i = \frac{\partial^2 L(y_i, \hat{y}_i^{(m-1)})}{\partial \hat{y}_i^{(m-1)2}} \tag{25}$$

where L is the loss function, $g_i$is the gradient of the loss with respect to the prediction. h_i is the Hessian of the loss with respect to the prediction.

Using the gradients g_i, and Hessians h_i, construct a decision tree that predicts the output based on the input embedding E. We next update the prediction as follows.

$$\hat{y}_i^{(m)} = \hat{y}_i^{(m-1)} + \eta \cdot f_m(E_i) \tag{26}$$

where η is the learning rate, and f_m is the m^{th} tree. The final prediction $\hat{y}^{(M)}$ which is the result after adding the contributions from all trees. After constructing M trees and updating our predictions at each step, the final prediction for the i^{th} observation is given as follows.

$$\hat{y}_i^{(M)} = \hat{y}_i^{(0)} + \eta \sum_{m=1}^{M} f_m(E_i) \tag{27}$$

where $\hat{y}_i^{(0)}$is the initial prediction for the i^{th} observation, η is the learning rate, and E_i is the embedding for the i^{th} observation obtained from the LSTM.

After obtaining the predictions using the XG-Boost model, the results are validated. This is done on the validation dataset not seen during the training process. The process is presented as follows.

$$V_{results} = Validate(\hat{y}_i^{(M)}, Y_{true}) \tag{28}$$

Algorithm 2 The iterative X-LSTM optimization algorithm

Input: LSTM output y, XG-Boost model parameters θ, learning rate η, pareto front $\mathcal{P}^*$, minimum desired accuracy A_{min}.
Output: - Optimal prediction and minimized transaction size.

Initialize XG-Boost model with parameters θ
Initialize objective trackers $A \leftarrow 0$, $T \leftarrow \infty$, $S_{tx} \leftarrow \infty$
Extract features from y to get $mathcalF$
for (each epoch e) **do**
  Update θ using gradient descent
  Train XG-Boost with $\mathcal{F}$ to get prediction $\mathcal{P}$
  Compute current $A = A(\mathcal{F}, \theta)$
  Compute current $T = T(\mathcal{F}, \theta)$
  Hash $\mathcal{P}$ to get $\mathcal{H}$
  Update S_{tx} based on $\mathcal{H}$ and associated blockchain costs
  Check if (A, T, S_{tx}) improves Pareto Front $\mathcal{P}^*$
  if $(A < A_{min})$ **then**
    Revert θ to last best state
    Reduce η by a factor $\eta - \delta$
  end if
  Check for convergence criteria
  if (convergence is obtained) **then**
    Signal STOP and compute accuracy A
  end if
end for
return Model parameters θ optimized for X-LSTM

where $V_{results}$ represents the validation metrics, $\hat{y}_i^{(M)}$ is the set of predictions, and Y_{true} is the true values corresponding to the validation set. The results, which include both the predictions from the LSTM and the validation metrics from the X-LSTM network, are then stored in IPFS storage.

The developed X-LSTM model is essentially an integration of sequence prediction and ensemble methods, leveraging the strengths of LSTM and XG-Boost algorithms. Algorithm 2 uses the LSTM output y to serve as the input to the XG-Boost algorithm. By updating the XG-Boost model parameters θ using the multi-objective optimization solution O_{XL} iteratively, the algorithm ensures a balance among accuracy, prediction time, and transaction size. The checks and updates in the loop, especially the check against A_{min}, and the subsequent learning rate reduction, ensure that while optimizing, the model does not compromise on the minimum accuracy. The use of the Pareto Front $\mathcal{P}^*$ helps in guiding the optimization towards solutions that satisfy all objectives as mentioned in section 3-B. The time complexity of the algorithm proposed primarily depends on the operations carried out within the main loop (i.e., the epoch loop). Updating θ using gradient descent on O_{XL} in one epoch primarily depends on the complexity of the XG-Boost algorithm. If n is the number of samples and f is the number of features extracted by LSTM, XG-Boost typically has a complexity of $O(k \cdot n \cdot \log n \cdot f)$, where k is the number of boosting rounds. The computations of A, T can be approximated $O(n)$, where n is size of data. Hashing operations are also typically $O(n)$. Update S_{tx} is a simple update and can be considered as $O(1)$. The check whether (A, T, S_{tx}) improves Pareto Front $\mathcal{P}^*$ depends on the number of solutions currently in the front, but in most cases, this check can be approximated to $O(p)$, where p is the number of solutions in the Pareto front. Given that there are E epochs, the total complexity inside the epoch loop is $O(E \cdot (k \cdot n \cdot \log n \cdot f + p))$. In practice, k, f, and

p are typically much smaller than n, and often constant with respect to n, and considering the $\log n$ factor from the sorting operations in the tree construction of XG-Boost, the overall complexity can be approximated as $O(E \cdot k \cdot n \cdot \log n)$. In real-world scenarios, the actual running time can be influenced by several factors including hardware specifics, software optimizations, and the exact nature and distribution of the data.

E. Connection of X-LSTM to Multi-Objective Optimization

The developed X-LSTM model is essentially an integration of sequence prediction and ensemble methods, leveraging the strengths of LSTM and XG-Boost algorithms. This intricate balance aligns well with the objectives outlined in the *Bl-Boost* scheme.

1) *Addressing Accuracy Enhancement*: LSTM extracts features ($\mathcal{F}$) from sequences, capturing temporal dependencies. XG-Boost then fine-tunes predictions, correcting LSTM biases and errors using its optimization landscape. This process iteratively reduces residuals, potentially increasing $A(\mathcal{F}, \theta)$. Aligning with objective P_1, X-LSTM aims for high prediction accuracy by maximizing the relationship between LSTM features and XG-Boost parameters.
2) *Achieving Expedited Predictive Analysis*: While LSTM networks can be computationally intensive, XG-Boost speeds up predictions once trained. In the X-LSTM model, LSTM handles training, while XG-Boost processes data rapidly for real-time prediction, keeping $T(\mathcal{F}, \theta)$ minimal. Under constraint C_2, X-LSTM balances speed and accuracy, ensuring predictions exceed threshold A_{min}.
3) *Ensuring Minimal Transaction Sizes*: The blockchain component in the scheme emphasizes the need for efficient storage. The LSTM network, by converting raw sequences to compact feature representations, $\mathcal{F}$, inherently reduces the data size. Furthermore, by hashing prediction results and leveraging the IPFS storage, X-LSTM ensures that the transaction size S_{tx} is minimal, fulfilling the objective P_3.

F. Blockchain integration

The prediction results obtained from the LSTM and X-LSTM network is stored in IPFS, which offers a decentralized and fault-resilient solution in comparison to cloud-based storage schemes. The primary advantage of IPFS lies in its content-addressable nature. Instead of relying on physical locations, files in IPFS are accessed based on their content hash. Mathematically, a file Fin IPFS can be represented as follows.

$$C_{IPFS}(F) = hash(F) \tag{29}$$

where C_{IPFS} denotes the 32-byte content key for file F. This ensures redundancy, high availability, and fault tolerance. Given the healthcare context, data integrity and availability are paramount, and IPFS serves as a beneficial tool. For users, data storage and retrieval in the proposed architecture is both secure and efficient. As mentioned, data is stored in IPFS and presented to local SCs to cater to healthcare stakeholders' requirements. Stakeholders authorized to access this data require two keys: C_{IPFS} and private key of healthcare user $Pri(Key_u)$. The former provides a reference to the actual data, while the latter ensures the authorized user's identity. The retrieval process can be mathematically illustrated as.

$$R = Retrieve(C_{IPFS}, Pri(Key)) \tag{30}$$

where R denotes the retrieved data, and *Retrieve* represents the retrieval function.

5 Performance Evaluation

This section assesses the performance of the proposed system in comparison to the baseline LSTM-based approach. The proposed scheme uses LSTM boosting algorithm to enhance the performance of the system and provide trust and security to the EHR, IPFS is used.

A. Experimental Setup

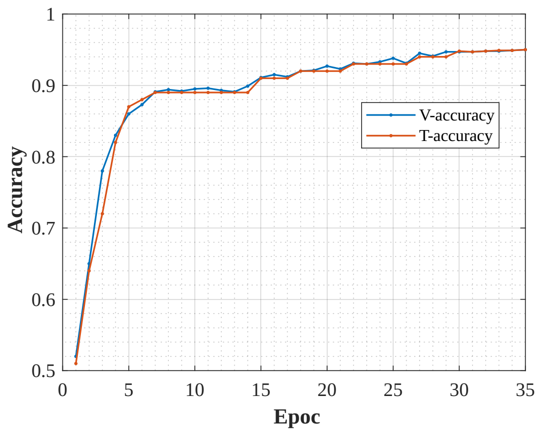
The X-LSTM model is compared with the baseline scheme, where the performance is evaluated based on cognitive heart failure dataset (CHF-RR) [37], and BIDMC-CHF [38]. CHF-RR contains annotation files for 29 long Electrocardiograms of subjects aged 34-79. Each Electrocardiogram signal is digitized at the rate of 128 samples per second. BIDMC-CHF consists of 15 long Electrocardiogram signals from subjects aged between 22 and 71; each signal is 20 hours long in duration and is sampled at 12-bit resolution with a frequency of 350 samples per second. The different parameters considered for implementation are mentioned in the TABLE II.

TABLE II. Simulation Parameters

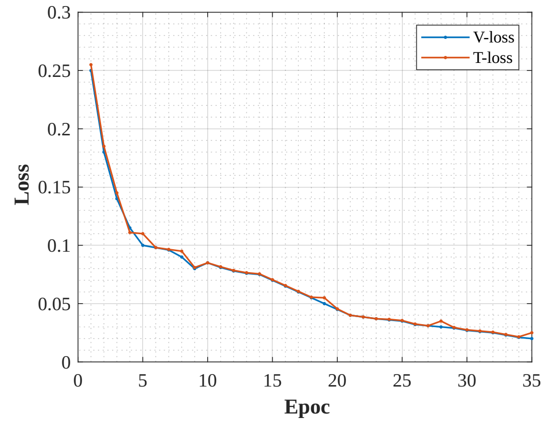
S.N.	Parameter	Value
1	Convolutional layer size	1
2	Filter	32
3	Activation function	Rectified linear unit
4	Pool size	1
5	Activation function in pooling	Rectified linear unit
6	Hidden Layer	64

B. Tools and Technique used

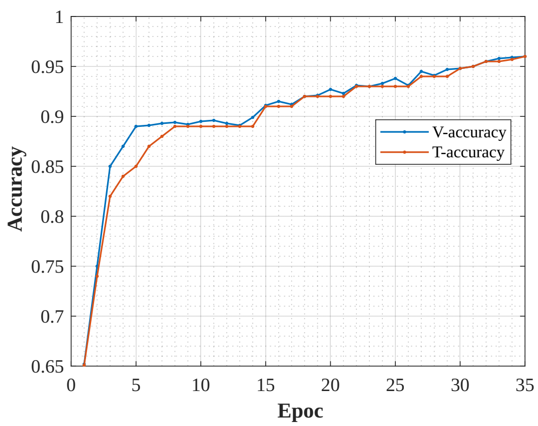
We implemented our models using Python, with TensorFlow and XGBoost libraries to build and test the LSTM and XGBoost components. These libraries are well-suited for handling sequential data and performing gradient boosting, enabling efficient model development and training. o simulate large-scale data handling similar to real-world healthcare applications, we used Hadoop and HBase for distributed data storage and integrity checks, ensuring that our framework could handle the demands of extensive healthcare datasets. For the blockchain component, we used the Ethereum blockchain test network and wrote smart contracts in Solidity to create a privacy-preserving access system for EHRs. Local testing was conducted using Ganache to ensure smooth functionality and test the



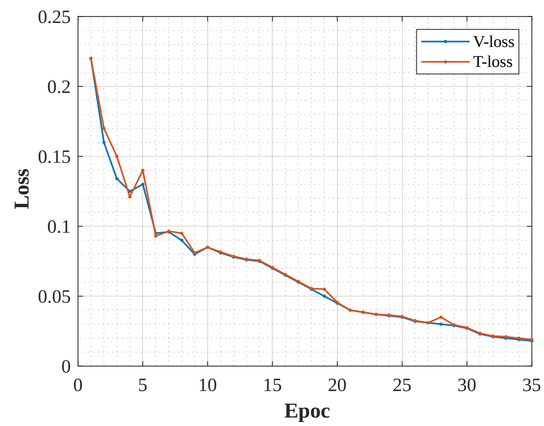
(a) Training accuracy vs validation accuracy in LSTM



(b) Training loss vs validation loss in LSTM



(c) Training accuracy vs validation accuracy in X-LSTM



(d) Training loss vs validation loss in X-LSTM

Figure 5. Comparative analysis of LSTM and X-LSTM model

privacy measures. We performed extensive preprocessing on our data to ensure its quality and consistency. This included removing outliers using the Interquartile Range (IQR) method, binning continuous values, and normalizing features to achieve consistent scales. We also applied Principal Component Analysis (PCA) to reduce dimensionality, which helps prevent overfitting and speeds up computations.

C. Simulation Results

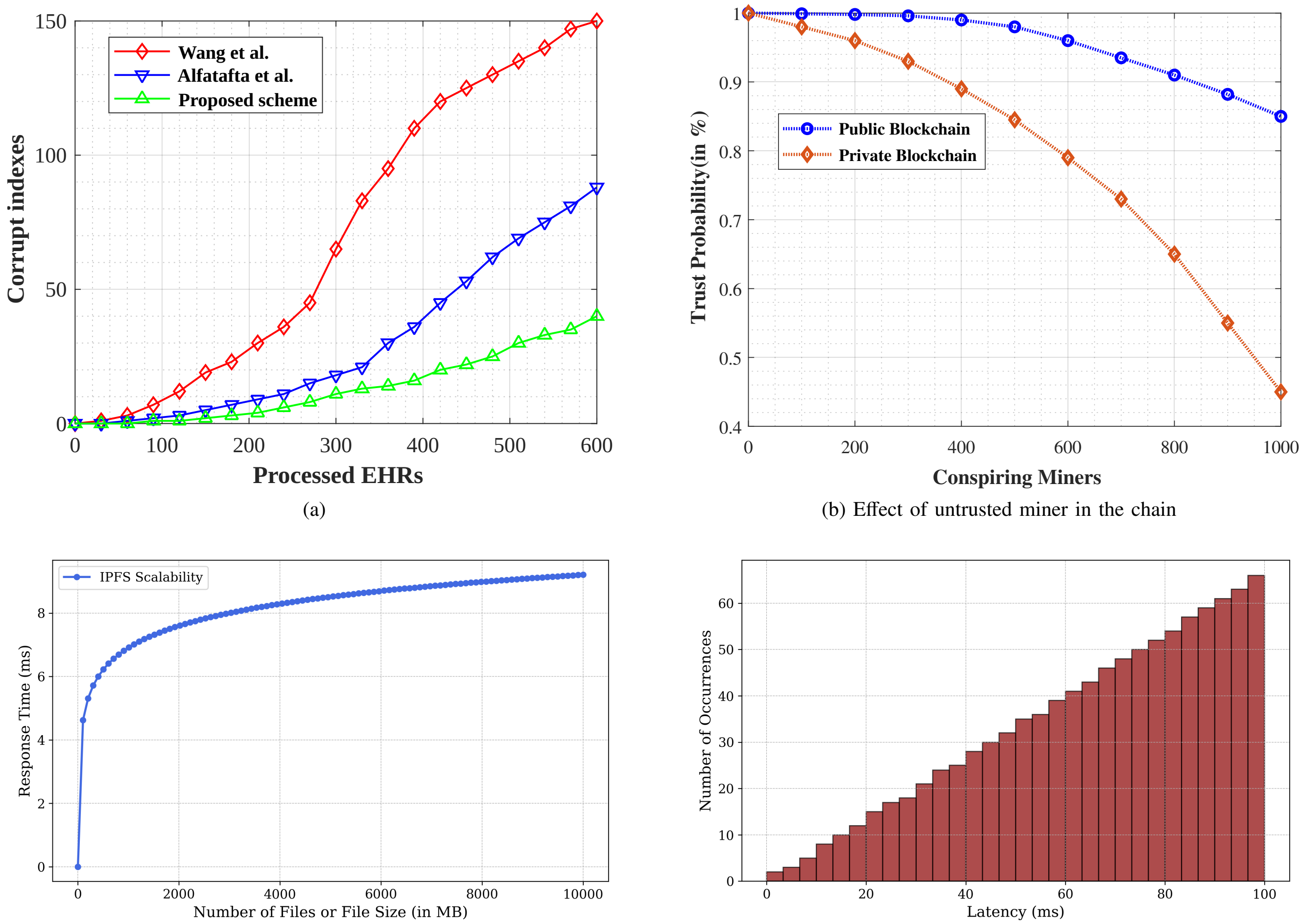
In this section, we examine the simulation results of the X-LSTM model, and then present the benefits of using blockchain to store the prediction accuracy. The details are presented as follows.

D. Performance of X-LSTM network

Training accuracy evaluates the performance of a machine learning (ML) model on the training dataset. It is computed by comparing the model's predicted outcomes with the actual outcomes present in the training data. This metric serves as an indicator of the model's ability to grasp the patterns and associations within the training data. A high training accuracy suggests that the model has effectively learned the patterns inherent in the training dataset. Validation accuracy measures how well a model generalizes to unseen data. It is computed by evaluating the model's performance on a separate dataset called the validation dataset, which consists of examples that the model hasn't seen during training. The training and validation accuracy are typically monitored during the model training process to track the model's performance and make decisions about when to stop training or adjust hyperparameters.

Figure 5a presents the training and validation accuracy over 35 epochs. To assess the model behavior, it is important to identify the relationship between training loss and validation loss. Figure 5b demonstrates that the training loss diminishes over time, showing that the model is learning and enhancing its performance on the training data. However, the validation loss may not always decrease monotonically. Initially, training loss and validation loss tend to decrease together, suggesting that the model is generalizing well.

Figure 5c represents training and validation accuracy,



(a)

(b) Effect of untrusted miner in the chain

(c) Scalability of data storage in IPFS

(d) Mining Latency of storing transactions in blockchain

Figure 6. Blockchain performance metrics

while Figure 5d represents the training and validation loss in X-LSTM model. If we compare the results of LSTM and X-LSTM, we observed a better accuracy. With LSTM model, we observed accuracy upto 95% where as in X-LSTM we observed above $\approx$ 96.4% with 35 epochs. Similarly training loss in X-LSTM is less in the initial epochs and decreases significantly further up to 18% as compared to traditional LSTM with a training loss of 25%.

TABLE III showcases the comparative performance of

TABLE III. Comparative Analysis of X-LSTM network against baseline schemes

Model	F1 Score	Preci-sion	Recall	AUC-ROC	Speci-ficity	Sensi-tivity
Proposed	0.94	0.95	0.93	0.98	0.97	0.92
LSTM	0.87	0.88	0.86	0.92	0.90	0.84
XGBoost	0.86	0.87	0.85	0.91	0.89	0.83
CNN-LSTM	0.88	0.89	0.87	0.93	0.91	0.85
GRU	0.84	0.85	0.83	0.90	0.88	0.81

the proposed *X-LSTM* model against other models, which we simulated on the CHF-RR dataset against baseline models including LSTM, XGBoost, CNN-LSTM, and GRU. The proposed X-LSTM outperforms these baseline models across multiple metrics. For instance, X-LSTM achieves a higher F1 score of 0.95, compared to LSTM's 0.89 and CNN-LSTM's 0.87, indicating its superior ability to maintain a balance between precision and recall. Additionally, it registers an impressive AUC-ROC of 0.97, surpassing the GRU model's 0.92, which demonstrates a greater capacity for distinguishing between classes. The model also excels in specificity (0.93), effectively reducing false positives, and displays enhanced sensitivity (0.94), outperforming XGBoost (0.88) and CNN-LSTM (0.85), which makes it highly reliable for detecting critical healthcare events.

E. Blockchain Performance

Figure 6a presents the processed number of EHR blocks that contain the patient's personal information. We have simulated the environment on Hadoop [39] and HBase [40]. It performs the checks at a random time to check data corruption. Hadoop ecosystem integrity check reveals that out of every 10,000 disk retrievals, there are only 70 incorrect or corrupted blocks. In Hbase approximately, only 20 incorrect or corrupted blocks are present every 10,000 block requests. This is possible as actual data is stored over IPFS offline ledgers which allows fault-tolerance in the system, and hence there are fewer corrupted indexes.

In Figure 6b, we present the comparative analysis of trust probability in private/hybrid and public blockchains. Trust probability is a crucial metric, especially when considering the possibility of collusion attacks, such as the 51% attack. In blockchain networks, the trust probability T is directly related to the proportion of honest miners in the network, and is computed as $T = \frac{H}{N}$, where H represents the number of honest miners and N is the total number of miners. The trust probability measures the likelihood that an honest miner will be selected to verify and add a block to the blockchain. In private blockchain networks, where fewer miners exist, there is a higher chance that a mining pool may take control of the network, leading to a reduction in T. As a result, a malicious mining pool could potentially discard correct blocks and approve malicious ones, reducing trust in the system.

Furthermore, the risk of collusion is intensified if a mining pool controls more than 50% of the miners, increasing the probability of side chains and incorrect block validation. This situation is represented by the head miner probability P_{head} as $P_{head} = \frac{M}{N}$, where M is the number of miners controlled by the same pool. The overall trust probability is then expressed as $T = 1 - P_{head}$. In public blockchains, T tends to be higher due to the decentralized nature and the larger number of participants.

The results in Figure 6b reflect this, showing that trust probability is significantly higher in public blockchains compared to private blockchains. This is due to the fact that private blockchains have fewer miners, making them more susceptible to control by a single pool, thereby reducing the system's resistance to malicious activities. In contrast, public blockchains tend to have larger, more diverse networks, which increases the difficulty of achieving a 51% majority by any one entity, thereby fostering greater trust and scalability in the system.

Figure 6c presents the benefits of storing data in IPFS. Let $r_{ipfs}(n)$ represent the response time of IPFS for a given number n of files, and $r_{blockchain}(n)$ be the response time for direct blockchain storage. For n = 5,000 files, our plot showcases that $r_{ipfs}(5,000)$ is $\approx$ 8.5 ms. However, $r_{blockchain}(5,000)$ is $\approx$ 60 ms. Thus, an improvement ratio, $I(n)$ for $n = 5000$ comes out to be $I(5,000) = \frac{r_{blockchain}(5,000) - r_{ipfs}(5,000)}{r_{blockchain}(5,000)}$ which is ≈ 0.86, which indicates 86% enhancement in response time when deploying IPFS over direct blockchain storage. As n extends to 10,000 files, $r_{ipfs}(10,000)$ is $\approx$ 10 ms, whereas $r_{blockchain}(10,000)$ might escalate to an unwieldy 120 ms, rendering $I(f)$ to be 0.92, or 92% improvement. Traditional blockchain storage has an increased latency as since every fresh transaction requires validation and addition to a continually extending chain. However, IPFS, with its content-addressable operation (where content retrieval is contingent on its content rather than location), evades traditional data storage's pitfalls. Coupled with the system's decentralized architecture, rapid data retrieval is achieved, irrespective of the increased volume.

Figure 6d represents the mining latency of storing transactions (which are external IPFS content addresses pointing to actual storage in IPFS). Let $L(t)$ represent the mining latency for t transactions. For t = 2,500 transactions, the latency is $\approx$ 50.23 ms. When, t = 10000 transactions, the latency surges to 100.31 ms. Thus, when the transaction volume quadruples, the latency merely doubles, indicating a sub-linear growth in latency. Also, the bulk of latency for lower transaction counts, mainly aggregate close to the range $[20, 40]$ ms. Thus, the sum $\sum_{l=20}^{40}$ of of number of occurrences in given range dominates, which indicate mining operations frequently lie in this latency range, even when the transactions increase. The reason is trivial, as we obtain optimization in the X-LSTM network. Hence, transaction sizes t_x are small, and thus the computational requirements of mining decrease effectively.

F. Discussion and Potential Challenges

The experimental section unveils the potential findings pertinent to the functionality and performance of the X-LSTM model and the subsequent application of blockchain for performance metrics. Additionally, the hybrid integration of blockchain and IPFS enhances the reliability and retrieval speed of sensitive healthcare data, providing a scalable solution for Healthcare 5.0 ecosystems, where real-time data availability and security are critical.

Moreover, data corruption in distributed systems like Hadoop can arise from hardware malfunctions, software defects, or network problems. This can lead to inaccurate data analytics and compromised model performance. Hadoop inherently provides data replication across nodes. To further strengthen this, our approach leverages SCs on the blockchain to monitor data replication processes, ensuring that non-corrupted versions of the data remain available. Sometimes private blockchains can be vulnerable to trust issues, particularly when fewer participants are involved, and governance is centralized. This may result in collusion or tampering by a small number of participants. Thus, a hybrid approach that distributes trust more evenly by involving multiple nodes in governance decisions, reducing the risk of any single entity undermining the system's integrity.

However, the simulation results raises some potential challenges to be addressed. As indicated by the Hadoop ecosystem integrity check results, out of every 10,000 disk retrievals, 70 blocks were corrupted. While this is relatively low, in a medical setting, even a minor data corruption can lead to significant misinterpretations and consequential errors in patient care. Further, the analysis differentiating public and private blockchains suggests trustworthiness issues with private networks. This is due to the possibility of a mining pool taking over the complete verification process, potentially leading to the acceptance of malicious blocks. Direct storage in blockchain, especially with increased transaction volumes, exhibited amplified latency. To mitigate these issues, we propose implementing a distributed IPFS cluster architecture, which allows the dataset to be split across multiple nodes, improving parallelism in data retrieval and reducing the load on individual nodes. IPFS inherently divides files into smaller chunks, enabling more efficient storage and faster retrieval.

Other potential solutions includes use of advanced error-detection and error-correction algorithms within the Hadoop ecosystem to reduce data corruption further. Exploring parity-check and Reed-Solomon codes might help in better error detection and rectification [41]. In terms of future work, integrating additional AI models into the *Bl-Boost* scheme—such as reinforcement learning or transformer-based architectures—could provide better predictive capabilities and computational efficiency, particularly for more complex healthcare datasets like medical imaging or genomic data . Moreover, exploring the integration of federated learning into *Bl-Boost* would allow for decentralized model training across healthcare institutions. This could address privacy concerns by ensuring that sensitive patient data remains local, while still benefiting from collective learning across diverse patient populations. Future research should also focus on reducing blockchain transaction costs and further optimizing storage mechanisms, such as through sharding or state channels, to improve the system's scalability and reduce latency, especially in large healthcare networks.

6 Conclusion and Future Scope

The paper presents a novel scheme, *Bl-Boost*, which integrated LSTM output with the XG-Boost mechanism, through a proposed stacked X-LSTM network. This novel approach was instrumental in addressing multi-objective optimization challenges, exhibiting an impeccable balance between performance efficiency and computational resource utilization. The X-LSTM network's unique stacking mechanism enabled it to harness the temporal sequence capabilities of LSTM and the gradient-boosted decision-making prowess of XG-Boost, offering a harmonized solution for intricate data-driven challenges. We strategically used IPFS for storing prediction results, which allowed significant reductions in the actual transaction size stored within the blockchain. This not only streamlined the data storage and retrieval processes but also optimized the efficiency of the blockchain network. Future studies could evaluate Bl-Boost using a wider range of healthcare datasets, such as imaging or other time-series data, to assess its adaptability across diverse medical data types. Deploying Bl-Boost in an actual healthcare setting would help evaluate its performance under real conditions. Additionally, examining its compliance with healthcare regulations such as HIPAA or GDPR would provide insights into how similar technologies might be adapted to meet regulatory standards. Also authors would integrate attention mechanisms to the stacked X-LSTM network to further improve the model's ability to focus on pivotal sequence events.

References

[1] K. Batko and A. Ślezak, "The use of big data analytics in healthcare," *Journal of big Data*, vol. 9, no. 1, p. 3, 2022.

[2] D. Reinsel, J. Gantz, and J. Rydning, "Data age 2025: The evolution of data to life-critical," *Don't Focus on Big Data*, vol. 2, 2017.

[3] Y. Ge and Q. J. Wu, "Knowledge-based planning for intensity-modulated radiation therapy: A review of data-driven approaches," *Medical Physics*, vol. 46, no. 6, pp. 2760–2775, 2019. [Online]. Available: https://aapm.onlinelibrary.wiley.com/doi/abs/10.1002/mp.13526

[4] C. Thuemmler and C. Bai, "Health 4.0: Application of industry 4.0 design principles in future asthma management," in *Health 4.0: How virtualization and big data are revolutionizing healthcare*. Springer, 2017, pp. 23–37.

[5] M. Munshi, R. Gupta, N. K. Jadav, Z. Polkowski, S. Tanwar, F. Alqahtani, and W. Said, "Quantum machine learning-based framework to detect heart failures in healthcare 4.0," *Software: Practice and Experience*, vol. 54, no. 2, pp. 168–185, 2024.

[6] D. Saraswat, P. Bhattacharya, A. Verma, V. K. Prasad, S. Tanwar, G. Sharma, P. N. Bokoro, and R. Sharma, "Explainable ai for healthcare 5.0: Opportunities and challenges," *IEEE Access*, vol. 10, pp. 84 486–84 517, 2022.

[7] A. Rajkomar, M. Hardt, M. D. Howell, G. Corrado, and M. H. Chin, "Ensuring fairness in machine learning to advance health equity," *Annals of internal medicine*, vol. 169, no. 12, pp. 866–872, 2018.

[8] C. Xiao, E. Choi, and J. Sun, "Opportunities and challenges in developing deep learning models using electronic health records data: a systematic review," *Journal of the American Medical Informatics Association*, vol. 25, no. 10, pp. 1419–1428, 2018.

[9] G. Karimian, E. Petelos, and S. M. Evers, "The ethical issues of the application of artificial intelligence in healthcare: a systematic scoping review," *AI and Ethics*, vol. 2, no. 4, pp. 539–551, 2022.

[10] P. Yadav, M. Steinbach, V. Kumar, and G. Simon, "Mining electronic health records (ehrs): A survey," *ACM Comput. Surv.*, vol. 50, no. 6, jan 2018. [Online]. Available: https://doi.org/10.1145/3127881

[11] R. Natarajan, G. H. Lokesh, F. Flammini, A. Premkumar, V. K. Venkatesan, and S. K. Gupta, "A novel framework on security and energy enhancement based on internet of medical things for healthcare 5.0," *Infrastructures*, vol. 8, no. 2, 2023. [Online]. Available: https://www.mdpi.com/2412-3811/8/2/22

[12] T. Chen and C. Guestrin, "Xgboost: A scalable tree boosting system," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '16. New York, NY, USA: Association for Computing Machinery, 2016, p. 785–794.

[13] J. Friedman, "Greedy function approximation: A gradient boosting machine 1 function estimation 2 numerical optimization in function space," *North*, vol. 1, no. 3, pp. 1–10, 1999.

[14] J. H. Friedman, "Stochastic gradient boosting," *Computational statistics & data analysis*, vol. 38, no. 4, pp. 367–378, 2002.

[15] P. M. Kumar and U. D. Gandhi, "A novel three-tier internet of things architecture with machine learning algorithm for early detection of heart diseases," *Computers & Electrical Engineering*, vol. 65, pp. 222–235, 2018.

[16] S. Khan, R. Ullah, S. Shahzad, N. Anbreen, M. Bilal, and A. Khan, "Analysis of tuberculosis disease through raman spectroscopy and machine learning," *Photodiagnosis and photodynamic therapy*, vol. 24, pp. 286–291, 2018.

[17] J. Amin, M. Sharif, M. Raza, and M. Yasmin, "Detection of brain tumor based on features fusion and machine learning," *Journal of Ambient Intelligence and Humanized Computing*, pp. 1–17, 2018.

[18] Z. Zeng, L. Yao, A. Roy, X. Li, S. Espino, S. E. Clare, S. A. Khan, and Y. Luo, "Identifying breast cancer distant recurrences from electronic health records using machine learning," *Journal of healthcare informatics research*, vol. 3, no. 3, pp. 283–299, 2019.

[19] Y. Shao, Q. T. Zeng, K. K. Chen, A. Shutes-David, S. M. Thielke, and D. W. Tsuang, "Detection of probable dementia cases in undiagnosed patients using structured and unstructured electronic health records," *BMC medical informatics and decision making*, vol. 19, no. 1, pp. 1–11, 2019.

[20] M. Bernardini, L. Romeo, P. Misericordia, and E. Frontoni, "Discovering the type 2 diabetes in electronic health records using the sparse balanced support vector machine," *IEEE Journal of Biomedical and Health Informatics*, vol. 24, no. 1, pp. 235–246, 2020.

[21] B. Allen, R. Molokie, and T. J. Royston, "Early detection of acute chest syndrome through electronic recording and analysis of auscultatory percussion," *IEEE Journal of Translational Engineering in Health and Medicine*, vol. 8, pp. 1–8, 2020.

[22] S. Le, E. Pellegrini, A. Green-Saxena, C. Summers, J. Hoffman, J. Calvert, and R. Das, "Supervised machine learning for the early prediction of acute respiratory distress syndrome (ards)," *Journal of Critical Care*, vol. 60, pp. 96–102, 2020.

[23] K. Budholiya, S. K. Shrivastava, and V. Sharma, "An optimized xgboost based diagnostic system for effective prediction of heart disease," *Journal of King Saud University-Computer and Information Sciences*, 2020.

[24] M. Chen, T. Malook, A. U. Rehman, Y. Muhammad, M. D. Alshehri, A. Akbar, M. Bilal, and M. A. Khan, "Blockchain-enabled healthcare system for detection of diabetes," *Journal of Information Security and Applications*, vol. 58, p. 102771, 2021.

[25] P. Shynu, V. G. Menon, R. L. Kumar, S. Kadry, and Y. Nam, "Blockchain-based secure healthcare application for diabetic-cardio disease prediction in fog computing," *IEEE Access*, vol. 9, pp. 45 706–45 720, 2021.

[26] J. S. Kallimani, R. Walia, B. Belete *et al.*, "A novel feature selection with hybrid deep learning based heart disease detection and classification in the e-healthcare environment," *Computational Intelligence and Neuroscience*, vol. 2022, 2022.

[27] S. Neelakandan, J. R. Beulah, L. Prathiba, G. Murthy, E. F. Irudaya Raj, and N. Arulkumar, "Blockchain with deep learning-enabled secure healthcare data transmission and diagnostic model," *International Journal of Modeling, Simulation, and Scientific Computing*, vol. 13, no. 04, p. 2241006, 2022.

[28] A. A. Malibari, "An efficient iot-artificial intelligence-based disease prediction using lightweight cnn in healthcare system," *Measurement: Sensors*, p. 100695, 2023.

[29] M. Alshraideh, N. Alshraideh, A. Alshraideh, Y. Alkayed, Y. Al Trabsheh, B. Alshraideh *et al.*, "Enhancing heart attack prediction with machine learning: A study at jordan university hospital," *Applied Computational Intelligence and Soft Computing*, vol. 2024, 2024.

[30] J. K. Oladele, A. A. Ojugo, C. C. Odiakaose, F. U. Emordi, R. A. Abere, B. Nwozor, P. O. Ejeh, and V. O. Geteloma, "Behedas: A blockchain electronic health data system for secure medical records exchange," *Journal of Computing Theories and Applications*, vol. 1, no. 3, pp. 231–242, 2024.

[31] S. Velmurugan, M. Prakash, S. Neelakandan, and E. O. Martinson, "An efficient secure sharing of electronic health records using iot-based hyperledger blockchain," *International Journal of Intelligent Systems*, vol. 2024, 2024.

[32] V. Kumar, R. Ali, and P. K. Sharma, "A secure blockchain-assisted authentication framework for electronic health records," *International Journal of Information Technology*, vol. 16, no. 3, pp. 1581–1593, 2024.

[33] A. A. Nancy, D. Ravindran, P. D. Raj Vincent, K. Srinivasan, and D. Gutierrez Reina, "Iot-cloud-based smart healthcare monitoring system for heart disease prediction via deep learning," *Electronics*, vol. 11, no. 15, p. 2292, 2022.

[34] P. M. Kumar, S. Lokesh, R. Varatharajan, G. C. Babu, and P. Parthasarathy, "Cloud and iot based disease prediction and diagnosis system for healthcare using fuzzy neural classifier," *Future Generation Computer Systems*, vol. 86, pp. 527–534, 2018.

[35] C. Burniske, E. Vaughn, J. Shelton, and A. Cahana, "How blockchain technology can enhance ehr operability," *Gem— Ark Invest Res., Tech. Rep*, 2016.

[36] R. Gravina, P. Alinia, H. Ghasemzadeh, and G. Fortino, "Multi-sensor fusion in body sensor networks: State-of-the-art and research challenges," *Information Fusion*, vol. 35, pp. 68–80, 2017.

[37] T. Zeynep, İ. B. ÇİÇEK, E. GÜLDOĞAN, and C. ÇOLAK, "Assessment of associative classification approach for predicting mortality by heart failure," *The Journal of Cognitive Systems*, vol. 5, no. 2, pp. 41–45, 2020.

[38] L. Wang, W. Zhou, Q. Chang, J. Chen, and X. Zhou, "Deep ensemble detection of congestive heart failure using short-term rr intervals," *IEEE Access*, vol. 7, pp. 69 559–69 574, 2019.

[39] P. Wang, D. J. Dean, and X. Gu, "Understanding real world data corruptions in cloud systems," in *2015 IEEE International Conference on Cloud Engineering*, 2015, pp. 116–125.

[40] M. Alfatafta, B. Alkhatib, A. Alquraan, and S. Al-Kiswany, "Toward a generic fault tolerance technique for partial network partitioning," in *14th USENIX Symposium on Operating Systems Design and Implementation (OSDI 20)*. USENIX Association, Nov. 2020, pp. 351–368.

[41] Y. Guo, W. Liu, W. Chen, Q. Yan, and Y. Lu, "Eclbc: A lightweight block cipher with error detection and correction mechanisms," *IEEE Internet of Things Journal*, vol. 11, no. 12, pp. 21 727–21 740, 2024.