**International Journal of Computing and Digital Systems**
ISSN (2210-142X)
Int. J. Com. Dig. Sys. , No. (Mon-20..)

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

# *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 University, Jaipur, Rajasthan, 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 Mon. 20, Revised Mon. 20, Accepted Mon. 20, Published Mon. 20*

**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-Boost*, which combines extreme gradient boosting (XG) with long short term memory (LSTM) model for making accurate predictions on EHR data. 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 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 present sample contracts for data sharing, and for blockchain metrics, we validate our performance of scalability, IPFS cost, and trust probability against collusion attacks. The proposed outcomes indicate 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, 5G communication, and security solutions to create a patient-centric model. Blockchain ensures secure, decentralized trust and promotes 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 isn't enough for Healthcare 5.0; effective 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, we propose a framework, *Bl-Boost*, 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 scheme is explained in further sections of the paper.

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

### A. Novelty

Recent healthcare analytics leverage ML and DL techniques for EHR insights and sensor-driven real-time patient monitoring. However, data scarcity and privacy concerns reduce prediction accuracy. Our framework addresses these challenges, offering a secure, trusted, and scalable solution. Combining XG-Boost and LSTM allows operation with small labeled datasets. Coupled with blockchain and smart contracts, we ensure transparent EHR access control, achieving decentralized privacy and high prediction accuracy.

### B. 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* framework. 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.

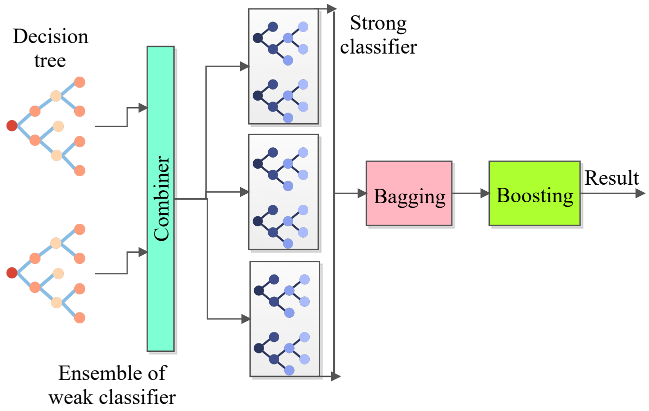


Figure 1. XG-Boost mechanism

### 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) 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 tracking [30]. 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 [31].

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

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. |
| 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

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 XG-boost based technique to improve the accuracy for disease detection and it perform 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 [32]. 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 ResNet-v2, and disease

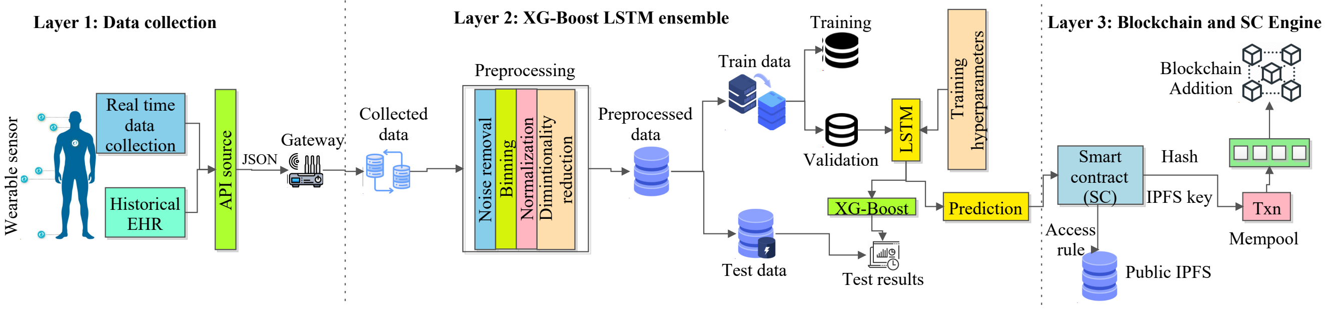


Figure 2. *Bl-Boost*: System Model

classification through a support vector machine (SVM), validated with benchmark medical images.

## 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 ($D_p$) comes from sensors like blood glucose, electrochemical, and amperometric biosensors. Data is processed using sensor fusion algorithms for uniform readings [33]. 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, and 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 set$D_{red}$.

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 $\theta$ (learning rate $\eta$, 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, $\theta$. 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 $\theta$ 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. Before venturing into the details of the X-LSTM approach, the data preprocessing steps are presented.

### A. Data Preprocessing

The collected data $D_c$ first undergoes for 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 $D_c$. 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 $D_{norm}$.

High-dimensional data can cause the curse of dimensionality and overfitting. Thus, we apply Principal Component Analysis (PCA), which finds orthogonal axes (principal components) that maximize data variance. The first few components capture most of the variance, allowing data projection onto this subspace. 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, $\lambda_i$ 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 makes it more computationally efficient for the subsequent LSTM processing, as it 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.

**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 $\Sigma$ of $D$
Compute the eigenvalues $\lambda$ and eigenvectors $v$ of $\Sigma$
Sort $\lambda$ in descending order and select the top $k$ eigenvectors to form matrix $P$
$D_{red} \leftarrow D \times P$
return $D_{red}$
```

### B. 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 3 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.

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

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

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

$$C_t = f_t \times C_{t-1} + i_t \times \tilde{C}_t \tag{16}$$

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \tag{17}$$

$$h_t = o_t \times \tanh(C_t) \tag{18}$$

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

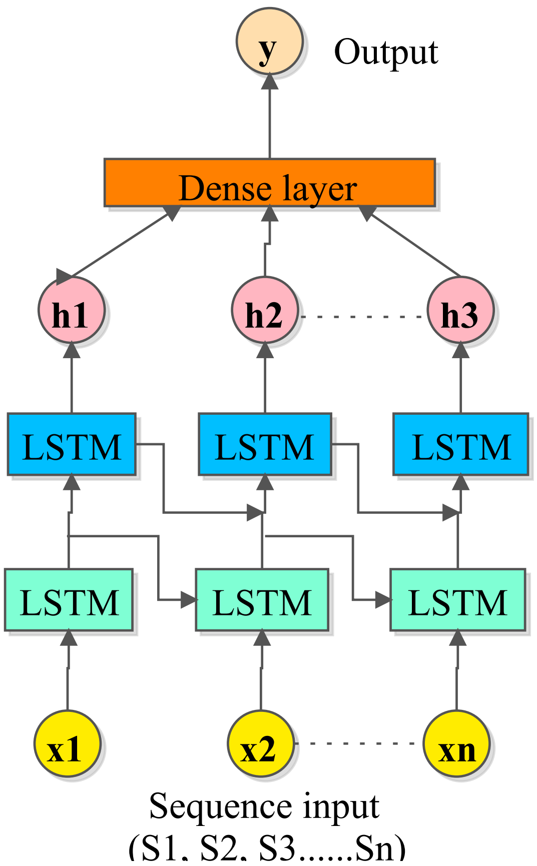


Figure 3. The stacked LSTM model

state, $[h_{t-1}, x_t]$ represents the concatenation of the previous hidden state and the current input, $o_t$ is the output gate, $\sigma$ 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.

### C. 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 $\eta$ 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, $\eta$ is the learning rate, and $E_i$ is the embedding for the $i^{th}$ observation obtained from the LSTM.

**Algorithm 2** The iterative X-LSTM optimization algorithm

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

```
Initialize XG-Boost model with parameters $\theta$
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 $\theta$ 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 $\theta$ to last best state
        Reduce $\eta$ 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 $\theta$ optimized for X-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.

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

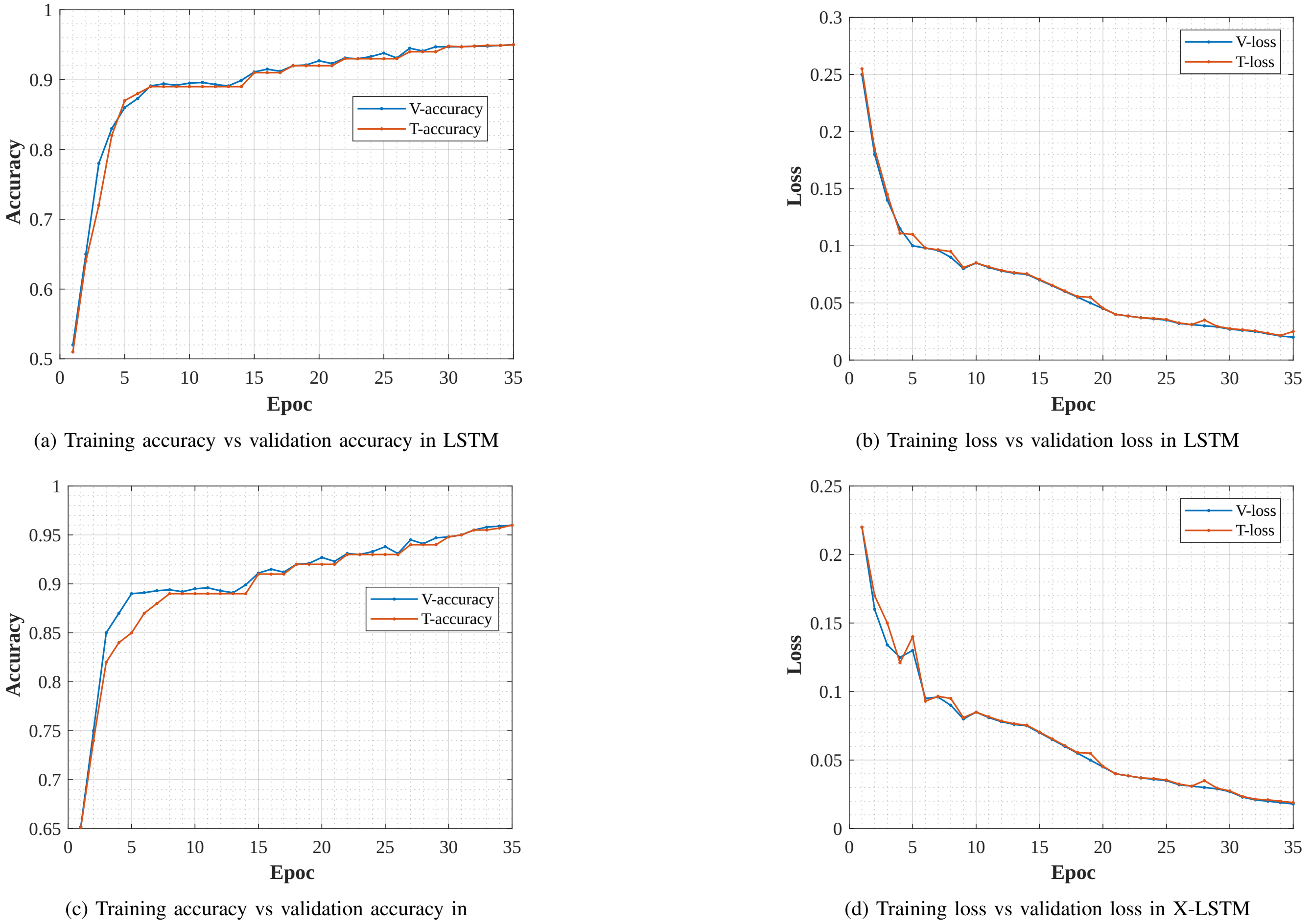
where $V_{results}$ represents the validation metrics, $\hat{y}_i^{(M)}$ is the set of predictions, and $Y_{true}$ is the true values corresponding to the validation set. The results, which include both the predictions from the LSTM and the validation metrics from the X-LSTM network, are then stored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

### *D. 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



(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 4. Comparative analysis of LSTM and X-LSTM model

storage, X-LSTM ensures that the transaction size $S_{tx}$ is minimal, fulfilling the objective $P_3$.

### *E. 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 $F$in 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) [34], and BIDMC-CHF [35]. CHF-RR contains annotations files for 29 long Electrocardiogram of subjects aged 34-79. Each Electrocardiogram signal is digitized at the rate of 128 samples per second. BIDMC-CHF 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 parameter considered for implementation is 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. 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.

### C. 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 4a 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 4b 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 4c represents training and validation accuracy, while Figure4d 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 ≈ 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%.

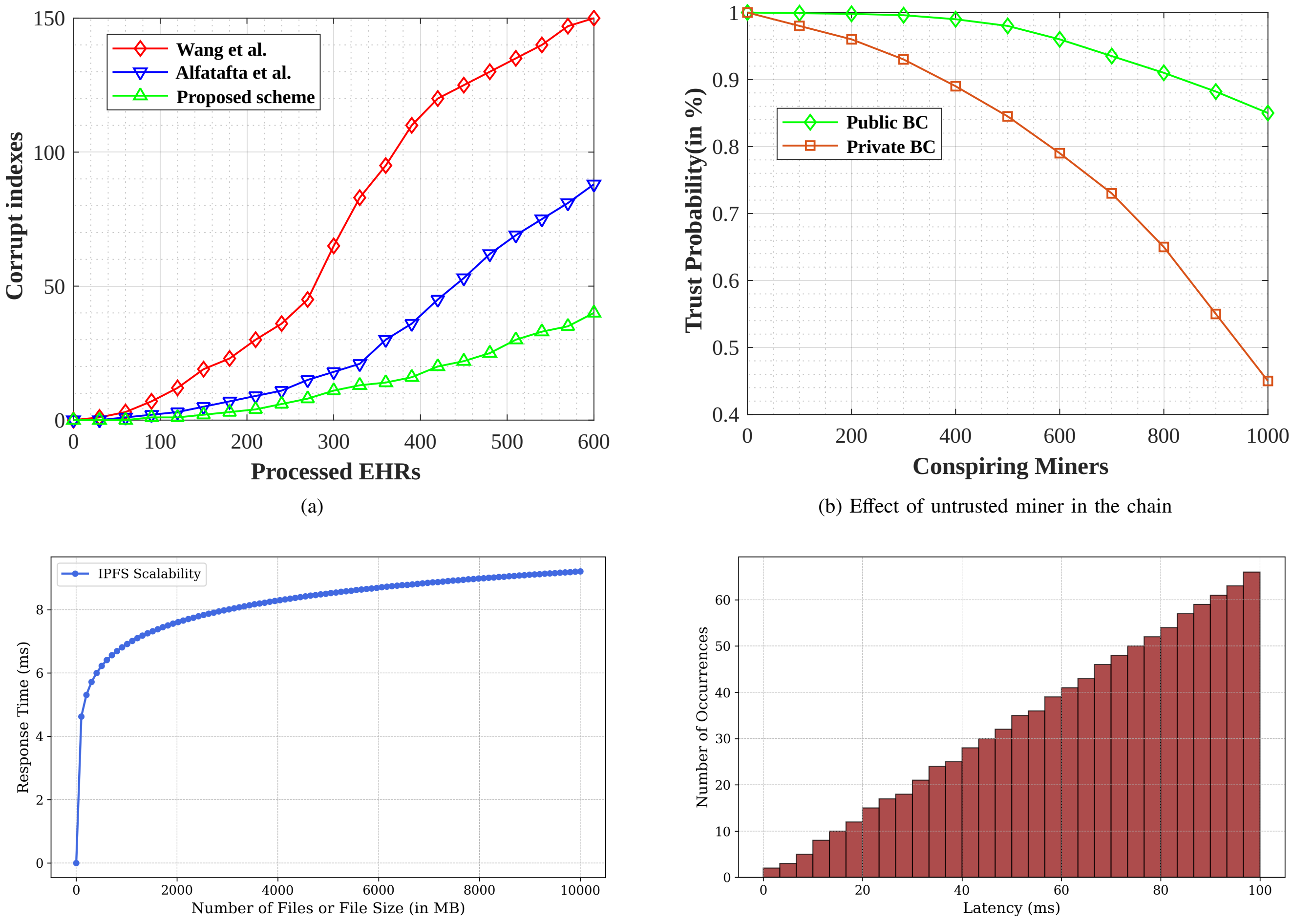
### D. Blockchain Performance

Figure 5a presents the processed number of EHR block that contains the patient's personal information. We have simulated the environment on Hadoop [36] and HBase [37]. It performs the checks at a random time to check data corruption. Hadoop ecosystem integrity check reveals that out of every 10,000 disk retrieval, there are only 70 incorrect or corrupted block. 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 5b we present the comparative analysis of private and public blockchain with trust probability which is measured on the basis of head miner, which is responsible for fair block addition. In private blockchain a mining pool can take over the the complete verification of block and can incorrectly discard the correct block and malicious one. In any mining network if we have more than 50% of the miner from the same pool they can grow side chain and discard the original one. Thus, public blockchain networks are more trusted. Public blockchains tend to have larger communities and user bases, which foster network effects. These network effects include greater liquidity, wider adoption, and more diverse applications and services built on top of the blockchain. Public blockchains also have the potential for interoperability, enabling different blockchains to communicate and share data. These aspects contribute to the overall growth and development of the ecosystem.

Figure 5c 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 ≈ 8.5 ms. However, $r_{blockchain}(5,000)$ is ≈ 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 ≈ 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 5d 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



(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 5. BC performance metrics

latency for $t$ transactions. For $t$ = 2,500 transactions, the latency is ≈ 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.

### *E. Discussion and Potential Challenges*

The experimental section unveils a range of salient findings pertinent to the functionality and performance of the X-LSTM model and the subsequent application of blockchain for performance metrics. Practically, the evident improvement of the X-LSTM model over the traditional LSTM—specifically a jump in accuracy to approximately 96.4% implies that the modifications incorporated are effectively capturing the intricacies of the Electrocardiogram signals in the datasets. Moreover, the utilization of blockchain technology to safeguard and validate data, especially in medical domain as critical as Electrocardiogram readings, underscores the potential of decentralized ledger technology in medical informatics. The added advantage of IPFS in enhancing data retrieval speeds is demonstrative of how modern distributed systems can revolutionize the storage and retrieval of patient data, making it quicker and more secure.

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. This delay can be problematic in real-time medical applications where instant data retrieval might be crucial.

As future scope, some potential solutions to address these challenges lie towards the need to integrate 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. To address the trust issues in private blockchains, hybrid blockchain architectures can be explored. Future research should delve deeper into optimizing the storage mechanisms in traditional blockchains. Utilizing sharding techniques or state channels might help reduce latency by partitioning the data and processing transactions off the main chain, respectively. Thus, while the results presented exhibit promise in the domain of medical data processing and storage using advanced algorithms and blockchain, there is significant room for improvements. The future lies in synergistically combining technological advancements with medical requirements.

## 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.

As part of future scope of the work, the authors would integrate attention mechanisms to the stacked X-LSTM network to further improve the model 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] 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.

[31] 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.

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

[33] 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.

[34] 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.

[35] 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.

[36] 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.

[37] 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.

**Varun Deshmukh** Varun Deshmukh received B.E. degree in Information Technology from Sri Sant Gajanan Maharaj College of Engineering Shegaon , SGB Amravati University , Amravati , Maharashtra in 2010 and M.Tech. from Shri Guru Gobind Singhji Institute of Engineering and Technology (SGGSIET), Nanded , Maharashtra in 2013. He is research scholar at Amity University Rajasthan, Jaipur. His research interests include Computer Network , Machine Learning , Business Intelligence.

**Sunil Pathak** is a distinguished Professor and Head of the Department of Computer Science and Engineering at Amity School of Engineering and Technology (ASET) in Jaipur, Rajasthan. He obtained his PhD from JK Lakshmipat University, Jaipur, in 2019. Prof. Pathak's research interests span a wide array of cutting-edge technologies, including Machine Learning, Health Informatics, Cloud Computing, Blockchain, and Wireless Communication. With a focus on innovation and academic excellence, Prof. Pathak is dedicated to shaping the future of technology through his research and teaching endeavors.

**Dr. Pronaya Bhattacharya** is an Associate Professor in the Department of Computer Science and Engineering at Amity School of Engineering and Technology, Amity University, Kolkata, West Bengal, India. He earned his PhD in Computer Science and Engineering specializing in Optical Networks from Dr. APJ Abdul Kalam Technical University, Lucknow. Prior to his doctoral studies, he completed his M.Tech in Information Technology with a focus on Network Security from Karnataka State University, Mysore, and his B.Tech in Computer Science and Engineering from Uttar Pradesh Technical University at Northern India Engineering College, Lucknow.His research contributions primarily revolve around the domain of Computer Science and Engineering, particularly in Optical Networks.