Frontiers in Health Informatics
ISSN-Online: 2676-7104
www.healthinformaticsjournal.com
2024; Vol 13: Issue 3
Open Access

Design and Development of Blockchain Enabled Smart Contracts - Enhancing with Hybrid Deep Learning Model

Etikala Aruna[1*], Arun Sahayadhas[2]

[1*]Research Scholar, Department of Computer Science and Engineering, VELS Institute of Science, Technology & Advanced Studies, Pallavaram, Chennai, India;
kattaaruna06@gmail.com (corresponding author)

[2]Professor, Department of Computer Science and Engineering, VELS Institute of Science, Technology & Advanced Studies, Pallavaram, Chennai, India. arun.se@velsuniv.ac.in

Cite this paper as: Etikala Aruna, Arun Sahayadhas (2024) Design and Development of Blockchain Enabled Smart Contracts - Enhancing with Hybrid Deep Learning Model. *Frontiers in Health Informatics, 13* (3), 4049-4067

ABSTRACT

Blockchain-enabled smart contracts have revolutionized secure, automated, and decentralized transaction handling across various industries. However, they face limitations in complex decision-making due to their rigid execution and predefined rules. This paper explores the integration of a hybrid deep learning model with blockchain-enabled smart contracts to enhance their functionality and decision-making capabilities. By embedding deep learning layers within the smart contract framework, this approach enables real-time data analysing, predictive analytics, and adaptive decision-making, fostering a more robust and dynamic contract execution. Through this integration, the hybrid model can analyze transaction data, external conditions, and contextual parameters, improving contract outcomes in applications such as finance, supply chain management, and healthcare. The study findings highlight the method's efficacy in enhancing smart contract flexibility and resilience while maintaining security and transparency.

Keyword: Blockchain, smart contracts, hybrid deep learning

1.INTRODUCTION

Health care is considered to be major concern for any nation for its overall growth and development [1-3]. Health care Industry has been revolutionized to health care 4.0 to provide much improvised diagnosis, disease treatment and even prediction of diseases. Health care4.0 incorporates the storage of the patient's records as electronic health records(EHR) which has been given an easy access to the doctors for a verification and diagnosis [4-5]. However, every person required to visit the hospital frequently for their check-ups and disease treatment.

This time consuming process has led to many death rates in recent times and this leads to implement the remote monitoring systems(RMS) in which the patients were deployed with wearable sensor and image devices, and the patients' real time health data [6] information are collected and stored as the electronic health records in the hospitals' huge servers which can be used for the further treatment and diagnosis. with an advent of artificial intelligence, remote health care systems have now able to analyse and interpret the larger number of medical

health records on their own for prediction of future diseases. Also these prediction results are used as the inputs to computers which can provide the disease diagnostic information to the doctors and patients. Many G-10 super powers such United States of America(USA), European Union(EU), Japan, China are concentrating on automated healthcare system with the help of artificial intelligence, to provide the quality health care solutions to the people. Hence the future remote health care systems will be intelligent, high accurate and personalized healthcare. This remains to be real challenge since the medical healthcare records contains high sensitive medical information which is required to be reliable and secured [7]. The introduction of blockchain is now mandatory to solve the above problem. Data stored on the blockchain is nearly tamper-proof, making unauthorized alterations or leaks of personal information highly unlikely. **This security feature has positioned blockchain as a leading technology in the modern smart healthcare market.**

Blockchains are carefully designed information advancements dependent on circulated figuring innovation. It is an innovation that stores information under administration in a circulated information stockpiling climate referred a "block", in which little information is associated in chain structure dependent on P2P technique, so nobody could adjust it discretionarily and anybody can see the consequences of alterations. The squares maintain a complete record of exchanges which was spread to clients before the square was found together with those that are shipped off all clients in a similar P2P way and therefore can't be changed or precluded self-assertively. Squares have a connection from the date of their identification and the past block, and a bunch of these squares is known as a blockchain. Basically, it is an innovation that assembles incalculable records in a group. Dissimilar to maintaining exchange documents on a focal worker, as on account of recently exchanged e-cash, blockchain shows exchange records to all clients and forestalls imitation by contrasting them and one another. It is evident that Bitcoin previously exhibited the idea of blockchain, and Ethereum initially executed the idea of a savvy money. There is a cozy connection among blockchain and cryptography. Blockchain isn't an innovation that must be utilized for encryption. Encoded cash is subordinate to blockchain. Along these lines, innovations or administrations previously applied with blockchain are being created.

A smart contract, or cryptocontract, is a self-executing program that automates the transfer of digital assets between parties based on pre-defined conditions [8-11]. Unlike traditional contracts, smart contracts enforce terms through code, without legal intermediaries, ensuring automatic execution as specified by the creator.

But still, medical data grows exponentially day by day, percentage of data to be saved in the block is constrained, it is essentially impossible to save every information in the blockchain. Hence the security and reliability of the medical health care data becomes nightmare for the real time implementation. **Thus blockchain enabled smart contract will be implemented.** Specifically, the research requires implementation for an effective authentication of large number of electronic health care records and need for applying the artificial intelligence blockchain enabled smart contracts over larger data [12-13].

Artificial Intelligence algorithms such as Deep Convolutional Neural Networks(DCNN), are utilized for the development of blockchain enabled smart contract technology which utilizes the above mentioned algorithm for the verification of the health records. Hence, this research proposes the new optimized deep learning blockchain model for an accurate prediction and verification of the EHRs which stores the medical data.

Contribution of the Research:

1. The research proposes the new methodology of integrating the gray wolf optimization technique over the deep convolutional neural network blockchain technology which can be used for better prediction.

2. The suggested model is juxtaposed with the different conventional artificial intelligence blockchain technology with the huge datasets.

3. To test the proposed framework, private artificial intelligence blockchain framework has been developed using the Web Flask framework with Python API packages.

This manuscript is organized in the following manner: Section-2 explores the relevant studies by multiple authors. The preliminary views of convolutional neural networks, grey wolf optimization and blockchain enabled smart contracts technology are discussed in Section-3. Section 4 outlines the working principle of the suggested architecture. The dataset descriptions, experimentations, along with an analytical discussion of the results are discussed in Section-5. Lastly, the study concludes with the future development in Section -6.

2.RELATED WORKS:

Li et al. (2024) [14] introduced ML2SC, a PyTorch to Solidity translator that enables the deployment of ML methods as smart contracts on blockchain networks. Their approach Utilizes a fixed-point arithmetic library to simulate floating-point operations, allowing for off-chain training and on-chain inference. The results demonstrated that gas costs increase linearly with MLP parameters, and the on-chain execution achieved identical classification accuracy to the original off-chain PyTorch execution.

Osei et al. (2024) [15] presented WIDENNET, a deep neural network-based method for identifying threats in smart contracts, specifically focusing on reentrancy and timestamp dependence vulnerabilities. Their approach converts bytecodes into Operational Codes (OPCODES) and transforms them into vector representations for neural network processing. The method achieved an average accuracy of 83.07% and precision of 83.13% when tested on real-world datasets, though it may require further optimization for different vulnerability types.

Tang et al. (2023) [16] proposed Lightning Cat, a DL -based solution for smart contract threat identification. Their approach used three optimized algorithms: CodeBERT, LSTM, and CNN. The Optimized-CodeBERT model demonstrated superior performance with an f1-score of 93.53% on the SolidiFI-benchmark dataset. The method successfully captured both syntax and code semantics through the CodeBERT pre-training method, though its effectiveness may vary with different vulnerability types.

Hasan (2023) [17] conducted comprehensive research on machine learning-based smart contract vulnerability detection, comparing multiple ML models including Decision Tree, Perceptron, SVM, and LSTM. Using data collected from etherscan.io, the study achieved notable results: 85.7% accuracy using SVM for the full dataset, 97.7% using LSTM for Reentrancy detection, and 100% accuracy for integer overflow/underflow detection using multiple models. While LSTM showed the highest overall accuracy, it had the highest computational cost.

Gopali et al. (2022) [18] employed artificial recurrent neural networks, specifically LSTM and Temporal Convolutional Network (TCN), to identify and classify vulnerable smart contracts into multiple categories: Suicidal, Prodigal, Greedy, and Normal. Their approach focused on addressing the continuous emergence of new attack vectors in smart contracts, though specific accuracy metrics were not provided.

Wu et al. (2022) [19] enhanced smart contract vulnerability detection by utilizing opcodes as static features and implementing bigram with Mtfidf optimization. Their comparative study of DL algorithms (CNN, LSTM, CNN-BiLSTM, and ResNets) demonstrated that ResNets achieved the best detection performance with 82% Macro-F1 score. The Mtfidf optimization significantly improved detection performance, though the approach may require regular updates to maintain effectiveness.

Cheng et al. (2021) [20] developed a reliable resource distribution framework utilizing smart contracts for blockchain-based IoT systems. Their solution incorporated a group-buying pricing framework and trust assessment system to handle resource pricing and assessment of service quality challenges. The simulation results showed improved transaction success rates and optimized user utility, though the implementation complexity may present adoption challenges.

3.BACKGROUND VIEWS:

This section details about the working mechanism of the convolutional neural networks, grey wolf optimizations.

3.1 Convolutional neural networks(CNN)

CNN represents a biologically inspired evolution of Multi-Layer Perceptron (MLP). CNNs are extensively employed for image classification, image clustering, and object detection in visuals. They are also utilized for optical character recognition and natural language processing. Besides images, when represented visually as a spectrogram, CNNs can additionally be applied to audio. Furthermore, CNNs have been directly applied to text analysis as well as in graph data using graph convolutional networks. The state-of-the-art proficiency of CNNs compared to their estimated algorithms contributes to their success in various domains.

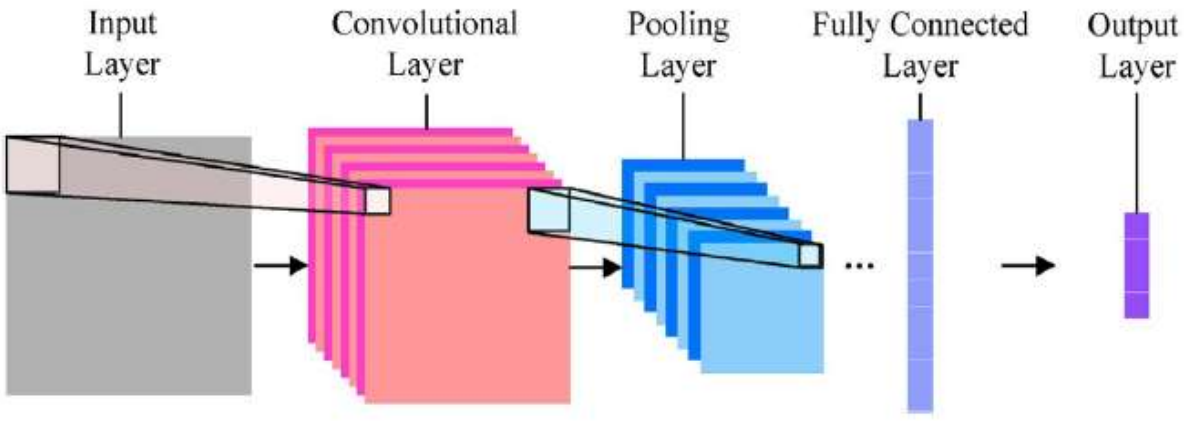

Figure 1 Graphical illustration of CNN.

As illustrated in Figure 1, in a Convolutional Neural Network (CNN), features are identified using filters, also known as kernels. A filter is merely a grid of attributes, referred to as weights, which are optimized for identifying unique characteristics. The use of the filter is to perform the convolution function, that involves an element-wise product and summation between two matrices. The training of the CNN is secured by minimizing the surplus of data in the input features. Consequently, he quantity of memory utilized by the network is likewise minimized. A prevalent method to accomplish this is max pooling, wherein a sliding window examines the input data, and the highest value within the window is pooled into an output matrix. The procedure is optimized for feature extraction by linking several convolutional layers and max pooling operations. The information is processed through these deep layers to generate feature maps, which are eventually transformed into a feature vector by passing through a Multi-Layer Perceptron (MLP). This is called a Fully Connected Layer, which executes high-level reasoning in the generated model.

If the k-th feature map at a specific layer is denoted as h^k, with its filters defined by the weights W^k and bias b^k, then the feature map hk is computed for the tanh activation function

$$h_{ij}^{k} = \tanh\left(\left(W^{k} * x\right)_{ij} + b^{k}\right) \quad (1)$$

The yield of the completely associated layer conveys probabilities of each class. The class conveying the most elevated likelihood is the arranged yield. The loads modification and improvement of the calculation is accomplished via the backpropagation of gradients.

3.2 GREY WOLF OPTIMIZATION:

In this segment, we introduce the proposed Grey Wolf Optimization (GWO) algorithm for the effective selection of predictive features. GWO is a nature-inspired meta-heuristic algorithm that emulates the predatory and social structure behaviours of grey wolves in the wild. The primary aim of this algorithm is to identify the most relevant features to improve prediction accuracy in a computationally efficient manner. Like other meta-heuristic algorithms, GWO seeks to balance exploration (searching for global solutions) and exploitation (refining local solutions) to find an optimal solution. In standard optimization techniques, maintaining this balance can be challenging, often leading to local optima or lengthy convergence times. GWO overcomes this by categorizing search agents into four roles: alpha, beta, delta, and omega wolves, where alpha wolves lead the exploration phase and adjust based on the beta and delta wolves' positions. This hierarchy allows for adaptive behaviour, enhancing the method's effectiveness to explore diverse options while maintaining stability during exploitation. By imitating these natural dynamics, GWO achieves robust performance in optimization tasks, efficiently converging towards optimal solutions while reducing computational overhead.

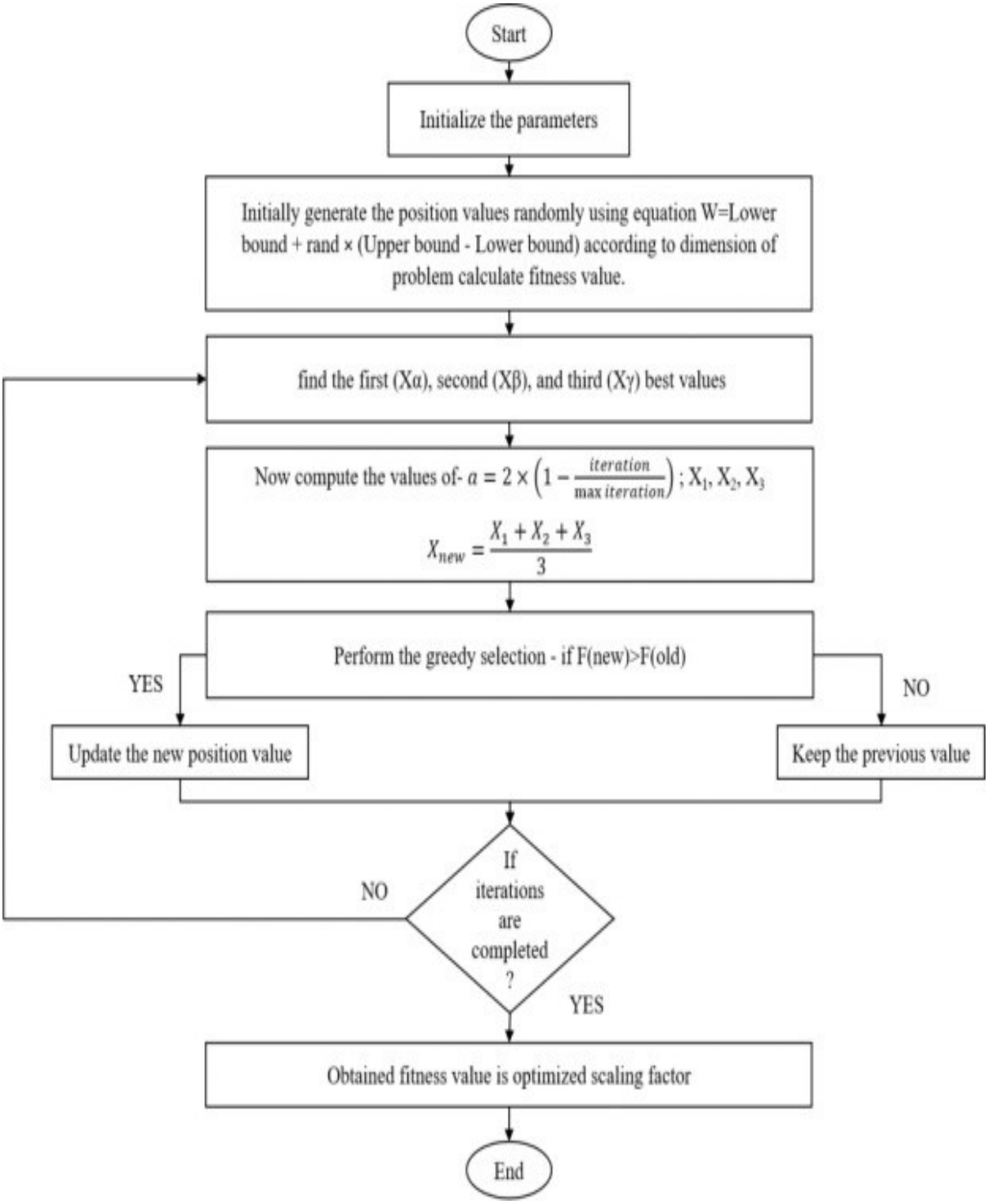

Figure 2 The main phases of the Grey wolf algorithm

3.2.1 PROCESS MECHANISM:

Step 1: Define the population size and cycles count and to begin the process we have to generate the starting values of wolves by using

W = lowerbound + rand * (upperbound – lowerbound)

Step 2: where W represents the initial values of the wolves, rand denotes the random function generating values between 0 and 1, and lower and upper bound indicate the range of parameters utilized in the equation that we need to optimize.

Step 3: After determining the initial values of the wolves, compute the function values. Then, based on these function values, obtain the first best position(Xa), second best(Xb) and third best (Xy).we have to proceed further by using Equations.

$$\begin{aligned} X1 &= X\alpha - A1 \times D \\ A1 &= 2 \times a \times rand - a \\ a &= 2 \times (1 - iteration/max.\ iteration) \\ C1 &= 2 \times rand \\ D &= |C1 \times X\alpha - X(t)| \end{aligned} \qquad (2)$$

Step 4 where X(t) indicates the current iteration, while **rand** signifies a random number ranging between 0 and 1, X1 is the value for first best A1 and C1 are coefficient vectors.

Step 5: Compute the updated position values of the wolves using the equation

$$X_{new} = (X1+X2+X3)/3 \qquad (3)$$

Step 6: where X_{new} displays the updated values of the wolves. Next, compute the value of the fitness function. Following that, identify the greedy solution to compare the new value with the previous one. If the new value is more optimized, replace it with the old value, and continue this process for the remaining iterations.

4.1 SYSTEM DESIGN:

Figure 4 shows the proposed architecture. This proposed system aims to estimate and assess the medical EHR utilizing the optimized DL and store it in the separate block for the further enhancement. The proposed systems consist of four different parts: 1. Data collection unit 2) Storing as Electronic Health Records (EHR)3. Optimized CNN models for the prediction and verification process 4. Block chain enabled smart contract components. This system has been designed to achieve the openness and safeness to manage and to distribute the medical data.

4.2 DATA COLLECTION UNIT:

By consolidating information from various sources, EHR datasets provide healthcare professionals and researchers with a holistic, data-driven approach to personalized patient care, predictive analytics, and medical

studies. This diverse, high-dimensional dataset serves as an ideal foundation for implementing machine learning models and blockchain technology to ensure secure, efficient, and accurate healthcare delivery

4.3 ELECTRONIC HEALTH RECORDS:

All the collected data were stored as general medical respiratory in the format which comprises demographics, comorbid conditions, specific laboratory data, and essential radiological observations.

PROPOSED RESEARCH METHODOLOGY:

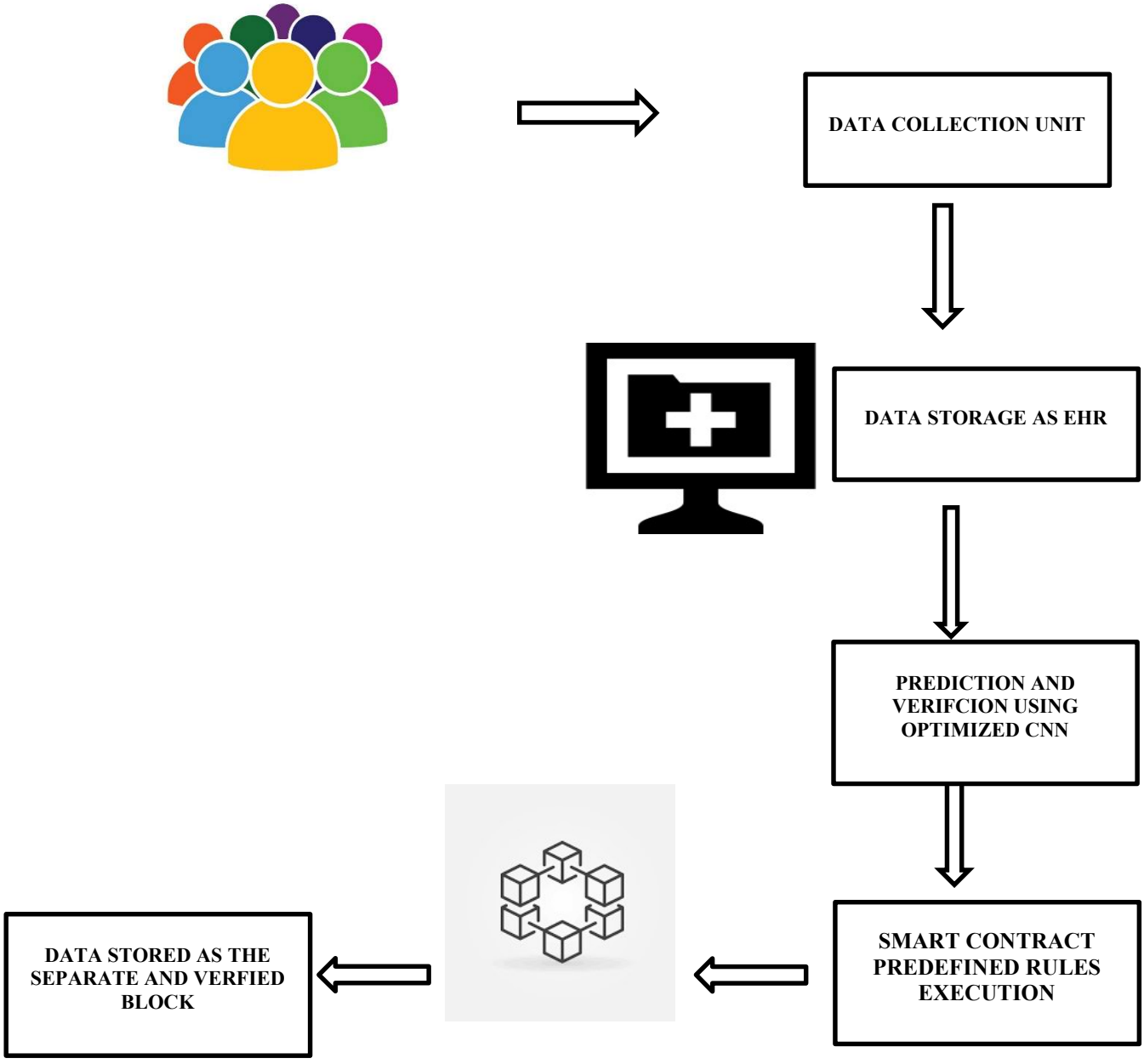

Figure 4 Working Mechanism for the Proposed Framework

4.3 OPTIMIZED CNN BASED PREDICTION AND VERIFICATION:

This section outlines the working mechanism of optimized convolutional neural networks (CNNs) for diagnostic prediction and data verification. As detailed in Section 3.1, CNNs are employed to predict diagnoses and perform classification. However, as dataset size increases, adding more convolutional layers can result in significant computational overhead and even lead to overfitting. To address these limitations, a computationally efficient model is required—one capable of accurate prediction and verification across diverse image datasets. This need has driven the development of a new hybrid learning model, designed to deliver high performance in both prediction accuracy and verification reliability.

4.4. PROPOSED OPTIMIZED CNN MODEL:

The Section 3.2 discusses the operational mechanism and benefits of grey wolf optimization. The detailed process for the optimized CNN layer is shown in Figure X. At first, a random set of bias weights and layers is provided to the convolutional networks, with no preset number of epochs to allow it to integrate into the deep learning network. The fitness function, based on prediction and verification accuracy, is established as a fixed criterion. During every iteration, input bias and layer configurations are measured utilising mathematical equations including an epoch count compatible with the learning network. These configurations are then fed into the convolutional training network, where the fitness function is evaluated. If the fitness function meets the threshold value, the process concludes; otherwise, it continues iteratively. The mathematical representation of the fitness function is outlined below.

$$Fitness\ Function = \ Max(Accuracy, Precision, Recall) \quad (4)$$

Sl.no	Algorithm-1 // Pseudo Code for the Optimized CNN//
01	Initialize the
	Total number of bees colonies g_m = *No of Layers/Weights*
	Leading constant Q ,
	Transfer intensityσ,
	Heuristic Factor β,
02	For i= 1 to G
03	Complete the Selection of the Bee nodes using Equation (2)
04	If K< K_{max}
05	Perform genetic operation such as Mutuation and Cross-over
06	Compute the Fitness Function utilizing Equation(4)
07	If fitness_function = Threshold
08	Proceed to Step 1
09	Else
10	Increment G and Go to Step 2

```
Else
        Perform the Cross-over and then Mutuation
            Compute the Fitness Function utilizing Equation(4)
If fitness_function = Threshold
        Proceed to  Step
Else
        Increment  G and Proceed to Step 2
End
End
End
End
```

4.5 OPTIMIZED CNN BLOCKCHAIN – SMART CONTRACT ARCHITECTURE:

The complete working of Blockchain Network is discussed in Section 3.3. proposed framework verifies the predicted rate with the original EHR. The verified data with the zero error are placed in each block of the blockchain. The proposed architecture uses the Hyper POR and blockchain agreement algorithm as mentioned. After placing in the blocks, smart contracts are created and then distributed separately by the consensus algorithm. The step by step working mechanism for the proposed architecture is given as

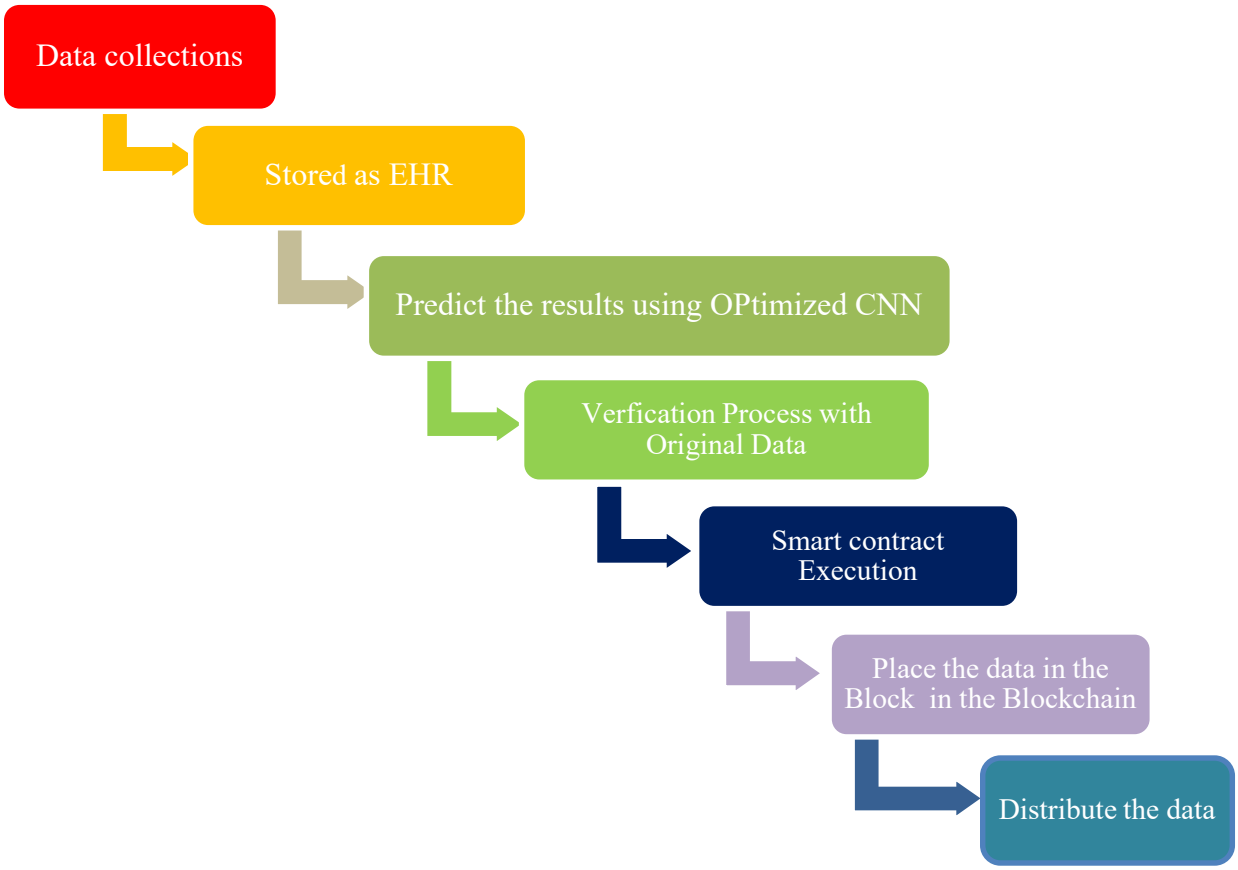

Figure 5 Complete Flow Diagram Illustrates the Working Mechanism of the Proposed Architecture

5.1 DATASET DESCRIPTION:

Electronic Health Records (EHR) datasets are comprehensive digital collections of patient health information, used widely in the healthcare industry to improve patient care and research. They typically include structured data, such as demographics, medical records, diagnoses, drug therapies, treatment regimens, immunization records, allergy information, radiological images, and laboratory results. EHRs offer a unified, longitudinal overview of patient information, enabling for consistent tracking of health trends over time. They also contain unstructured data, like physician notes, which, when processed with advanced tools, can yield valuable insights.

5.2 EXPERIMENTATION:

The collected datasets are compiled and stored as Electronic Health Records (EHR), which serve as input for the proposed architecture implemented using Python with a Flask API framework, running on the local server at 127.0.0.1:5000. Blocks for prediction and verification are generated following these specifications and operate on the local server at 127.0.0.1:8080. Additionally, 70% of the data collected is allocated for training, 20% for testing, and 10% for verification and validation.

$$\text{Accuracy} = \frac{\text{DR}}{\text{TNI}} \text{ x100} \quad (5)$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP+FN}}\text{x100} \quad (6)$$

$$\text{Precision} = \frac{\text{TN}}{\text{TP+FP}}\text{x100} \quad (7)$$

$$\text{F1-score} = 2.\frac{Precison*Recall}{Precision+Recal} \quad (8)$$

Where TP and TN denote True Positive and True Negative values, respectively, and DR and TNI represent the count of identified Results and the Total count of Iterations. The images are preprocessed before storing as the EHR format. Figure7 shows the block of the private blockchain created for storing the predicted and verified results of each individuals.

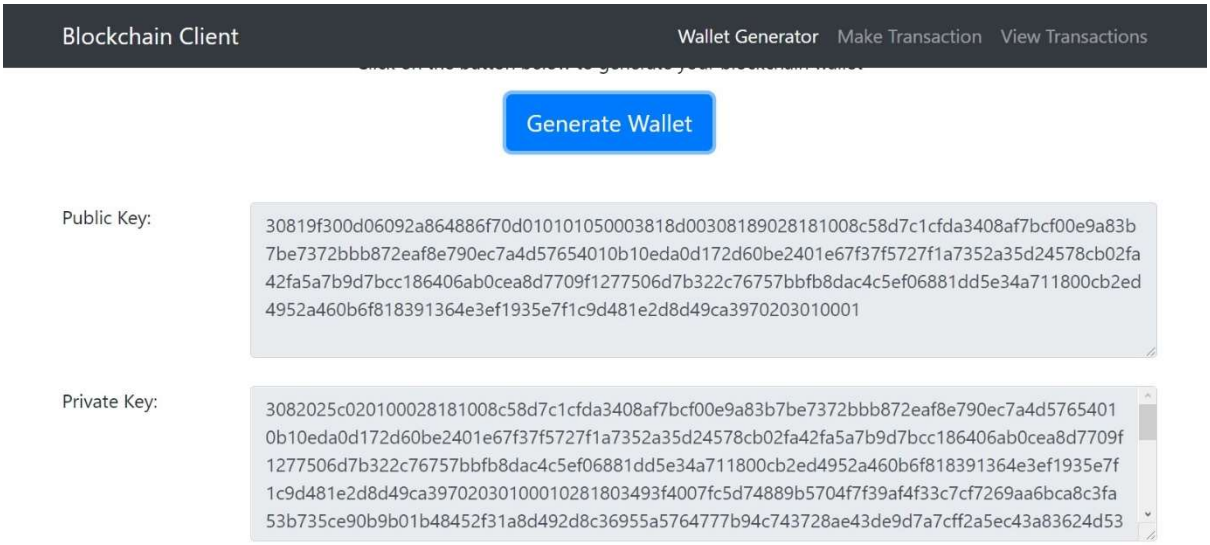

Figure 7 Block Created for Storing the Verified and Predicted Results of the individuals

5.3 FINDINGS AND ANALYSIS:

To evaluate the effectiveness of the suggested model, we have used the different epochs with learning rate of 0.001. It was determined that the optimal outcomes during the tuning process were achieved with 120 epochs, a learning rate of 0.001, and an output batch size configured to 170.

Table I Training Accuracy Performance using the no of batches =170

Sl.no	No of batches	No of Epochs	Training Accuracy (%)
01	170	40	96.45%
02	170	80	97.7%
03	170	120	98.6%
04	170	160	98.5%
05	170	200	98.2%
06	170	240	98.4%
07	170	280	98.30%

Table II Training Accuracy Performance using the no of batches =170

Sl.no	No of batches	No of Epochs	Testing Accuracy (%)
01	170	40	96.35%
02	170	80	97.64%
03	170	120	98.55%
04	170	160	98.36%
05	170	200	98.2%
06	170	240	98.35%
07	170	280	98.28%

Table III Validation/Verification Accuracy Performance using the no of batches =170

Sl.no	No of batches	No of Epochs	Validation /Verification Accuracy (%)
01	170	40	96.35%
02	170	80	97.64%
03	170	120	98.55%
04	170	160	98.36%
05	170	200	98.2%
06	170	240	98.35%
07	170	280	98.28%

Table IV Performance Metrics Evaluation for the Proposed Architecture using Testing Datasets

Sl.no	No of batches	No of Epochs	Precision (%)	Recall(%)	F1-Score
01	170	40	96.5%	96.%	96.34%
02	170	80	96.45%	96.3%	96.30%
03	170	120	97.65%	97.4%	97.5%
04	170	160	97.5%	97.1%	97.3%
05	170	200	96.45%	96.30%	96.20%
06	170	240	96.30%	96.20%	96.10%
07	170	280	96.20%	96.10%	96.18%

Table V Performance Metrics Evaluation for the Proposed Architecture using Verification/Validation Datasets

Sl.no	No of batches	No of Epochs	Precision (%)	Recall(%)	F1-Score
01	170	40	96.5%	96.%	96.34%
02	170	80	96.45%	96.3%	96.30%
03	170	120	97.65%	97.4%	97.5%
04	170	160	97.5%	97.1%	97.3%
05	170	200	96.45%	96.30%	96.20%
06	170	240	96.30%	96.20%	96.10%
07	170	280	96.20%	96.10%	96.18%

Table I illustrates the training accuracy achieved by the proposed architecture, reaching 98.6% at 120 epochs, while training accuracy across different epochs ranges from 96.45% to 98.30%. Therefore, the model is optimized at 120 epochs to achieve maximum training accuracy. Tables present the testing and verification accuracy, respectively, with the highest values observed at 98.55% for 120 epochs, confirming that this configuration maximizes performance. Precision, recall, and F1-score for the optimized 120 epochs are 97.65%, 97.4%, and 97.5%, respectively. Additionally, the effectiveness of the suggested framework is contrasted with existing deep learning blockchain architectures, including CNN blockchain frameworks, Backpropagation, and artificial neural blockchain models.

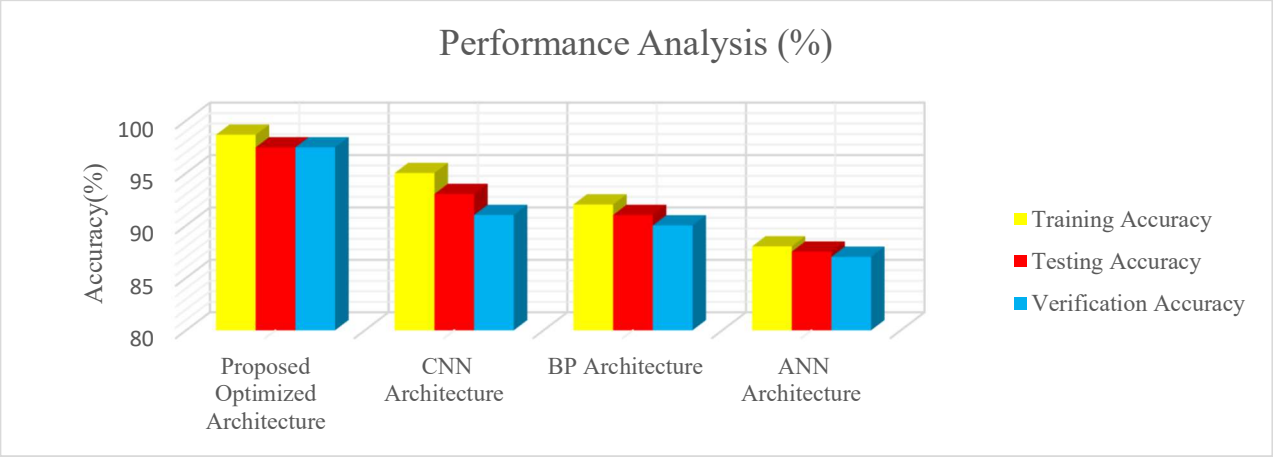

Figure 7 Comparative Assessment Between the suggested Framework with existing Learning Blockchain Frameworks.

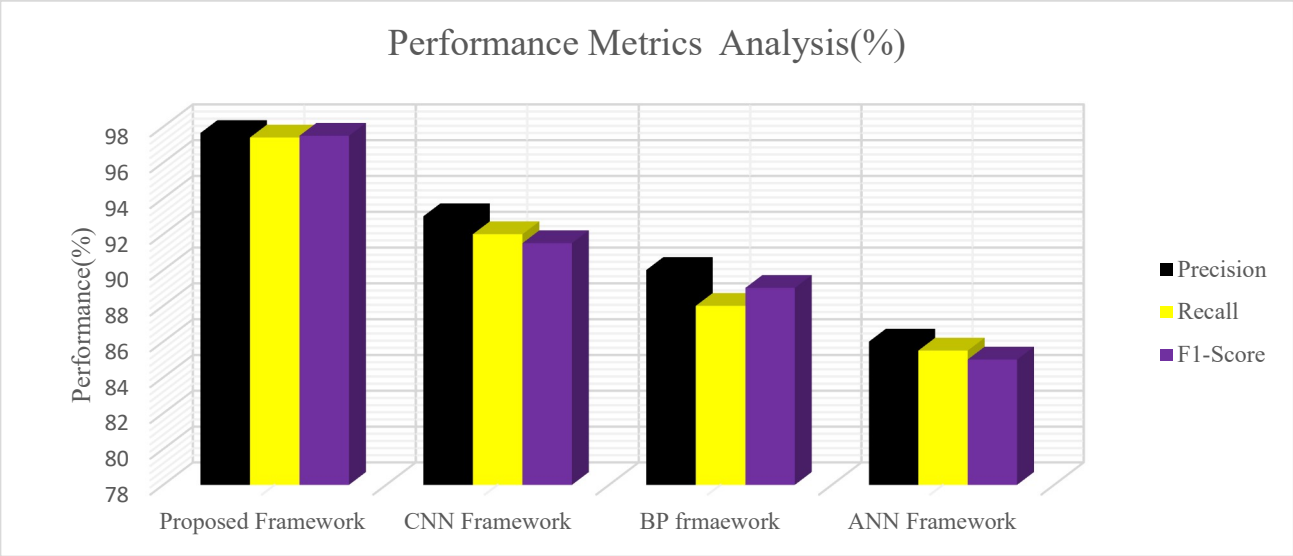

Figure 8 Comparative Assessment Between the Proposed Framework with Existing Learning Blockchain Frameworks.

Figure 7 illustrates the comparative assessment of accuracy between the suggested framework and other existing framework. It is evident that the suggested framework has yielded training accuracy as 98.5%, testing accuracy as 98% and verification accuracy as 97.45% which is 2% greater than CNN framework, nearly3-5% greater than BP -DL framework and 10% greater than ANN framework. It is also evident that the suggested framework has surpassed than the conventional frameworks and makes it suitable for the prediction and verification of different image data. The similar fashion is found in Figure 8 which illustrates the comparative assessment of different efficiency between the different frameworks. Furthermore, we have analyzed the impact of proposed framework on the blockchain architecture by using the following mathematical expressions

$$\text{Transaction per second(TPS)} = \frac{No\ of\ Transaction\ Processed}{No\ of\ transaction\ stored\ per\ \text{sec}} \quad (9)$$

The above two TPS and RT indicates the speed of the blockchain architecture. These parameters are used to indicate the speed in which TPS and RT was evaluated for the proposed artificial intelligence architecture. These performance tests are then compared with the other existing artificial intelligence blockchain architecture. To conduct this performance test, 100 nodes are created as blockchain clients with the different proxy addresses which ranges from 127.0.0.8080 to 127.0.0.8180. The transaction is enabled from the each node for accessing the data and various TPS are calculated.

Figure9 shows the TPS analysis between the proposed framework and blockchain framework without the artificial intelligence. It is clear that the number of TPS increases by the average of 300 TPS for 100 nodes. In addition, proposed blockchain 'consensus algorithm is compared with the other neural network based blockchain consensus algorithm which is shown in Figure10. It is found that the average TPS is 300 TPS, illustrating 100-TPS rise than CNN framework,250 TPS rise than BP-DL framework and even 600 TPS rise than ANN blockchain Framework.

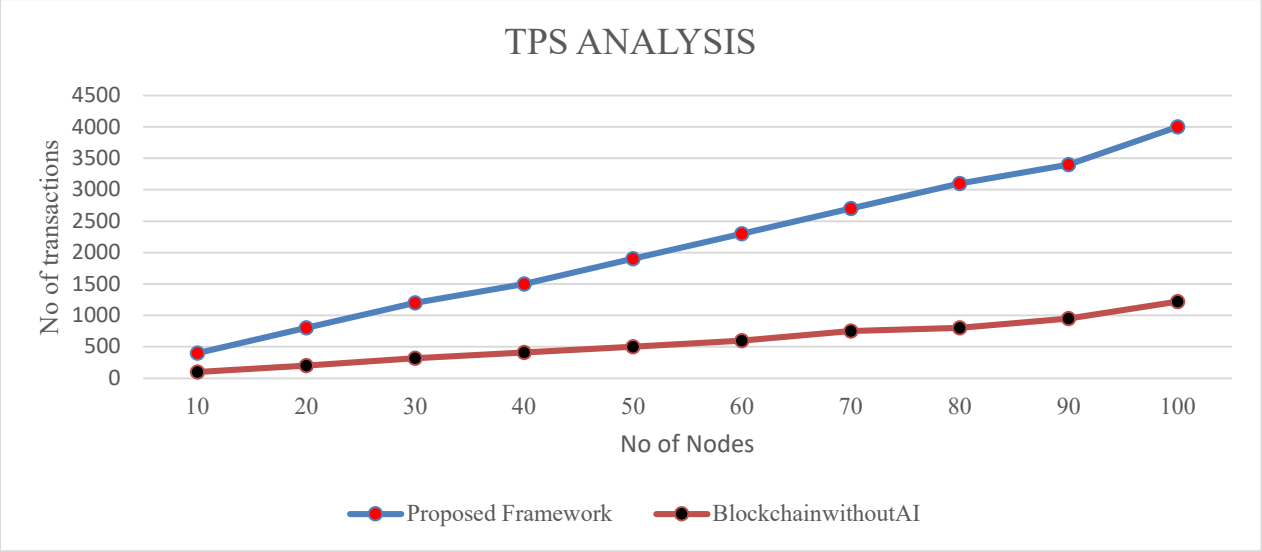

Figure 9 TPS Analysis for the Proposed Framework and Blockchain without AI System

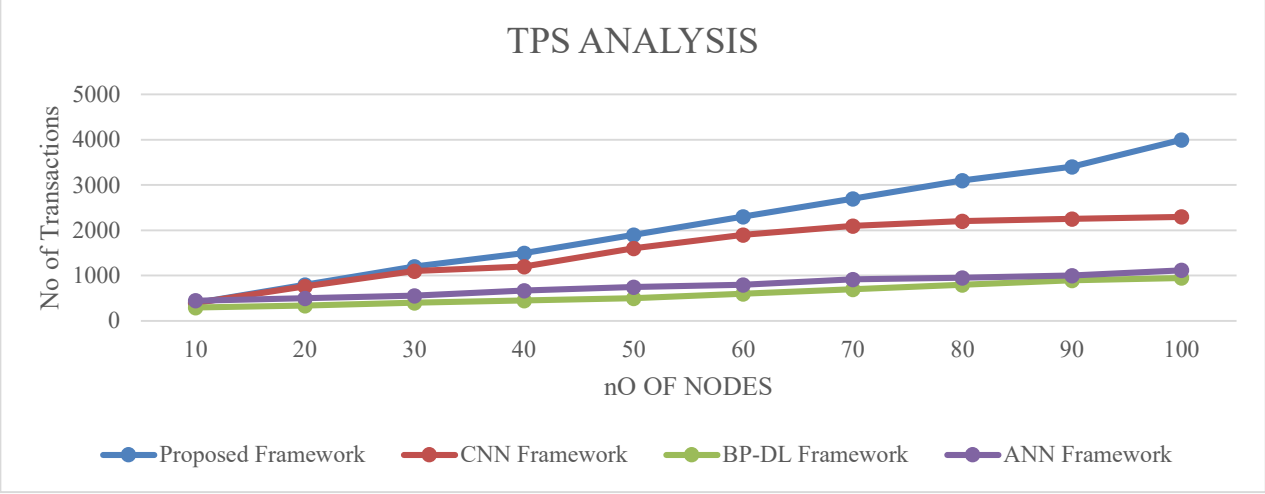

Figure10 TPS Analysis for the Different Artificial Intelligence Blockchain Framework with the increased number of nodes.

From the above figures, it is evident that the integration of optimized DL algorithms in the blockchain has increased the diagnosis rate, security and even speed of the blockchain

6.CONCLUSION AND FUTURE DEVELOPMENT:

Medical information systems today require thorough verification before administering diagnoses and treatments. With blockchain technology, data security has improved through immutability, yet real-time monitoring still demands data verification. This paper proposes a blockchain framework enhanced by a hybrid deep learning algorithm. Verified data blocks containing predicted diagnoses are created and distributed securely to hospitals and users. This framework addresses blockchain challenges like scalability, security, and handling large data. Tested using Python 3.8 and a Flask API on a private blockchain, the framework achieved 98% accuracy in diagnosis prediction and verification, outperforming existing algorithms. While AI-integrated blockchain technology is promising, further improvements are needed to ensure security and manage complex, heterogeneous image data for practical, safe implementation.

Data Availability

The data used to support the findings of this study, which includes a newly created dataset, is available from the corresponding author upon request.

Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this paper.

Funding Statement

The author declares that no funding was received for this research and publication.

REFERENCES:

[1] J. J. Hathaliya, S. Tanwar, S. Tyagi, and N. Kumar, “Securing electronics healthcare records in healthcare 4.0 : A biometric-based approach,” Computers & Electrical Engineering, vol. 76, pp. 398 – 410, 2019. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S004579061930062X .

[2] R. Gupta, S. Tanwar, S.Tyagi, N.Kumar, M.Obaidat, and B.Sadoun, “Habits: Blockchain-based telesurgery framework for healthcare 4.0,” IEEE International Conference on Computer, Information and Telecommunication Systems, pp. 1–5, 2019.

[3] Abid Haleem, Mohd Javaid, Ravi Pratap Singh, Rajiv Suman, Shanay Rab. Blockchain technology applications in healthcare: An overview.International Journal of Intelligent Networks, Volume 2, 2021, Pages 130-139, ISSN 2666-6030.https://doi.org/10.1016/j.ijin.2021.09.005.

[4] Han Y, Zhang Y, Vermund SH. Blockchain Technology for Electronic Health Records. Int J Environ Res Public Health. 2022 Nov 24;19(23):15577. doi: 10.3390/ijerph192315577. PMID: 36497654; PMCID: PMC9739765.

[5] A. Shahnaz, U. Qamar and A. Khalid, "Using Blockchain for Electronic Health Records," in *IEEE Access*, vol. 7, pp. 147782-147795, 2019, doi: 10.1109/ACCESS.2019.2946373.

[6] P. Kalpana, P. Srilatha, G. S. Krishna, A. Alkhayyat and D. Mazumder, "Denial of Service (DoS) Attack Detection Using Feed Forward Neural Network in Cloud Environment," *2024 International Conference on Data Science and Network Security (ICDSNS)*, Tiptur, India, 2024, pp. 1-4, https://doi.org/10.1109/ICDSNS62112.2024.10691181.

[7] Mandarino V, Pappalardo G, Tramontana E. A Blockchain-Based Electronic Health Record (EHR) System for Edge Computing Enhancing Security and Cost Efficiency. *Computers*. 2024; 13(6):132. https://doi.org/10.3390/computers13060132

[8] P. Kalpana, K. Malleboina, M. Nikhitha, P. Saikiran and S. N. Kumar, "Predicting Cyberbullying on Social Media in the Big Data Era Using Machine Learning Algorithm," *2024 International Conference on Data Science and Network Security (ICDSNS)*, Tiptur, India, 2024, pp. 1-7, https://doi.org/10.1109/ICDSNS62112.2024.10691297.

[9] Nabi, S. A., Kalpana, P., Chandra, N. S., Smitha, L., Naresh, K., Ezugwu, A. E., & Abualigah, L. (2024). Distributed private preserving learning based chaotic encryption framework for cognitive healthcare IoT systems. *Informatics in Medicine Unlocked*, *49*, 101547. https://doi.org/10.1016/j.imu.2024.101547

[10] Mohanta, Bhabendu & Panda, Soumyashree & Jena, Debasish. (2018). An Overview of Smart Contract and Use Cases in Blockchain Technology. 10.1109/ICCCNT.2018.8494045.

[11] Zheng X. Research on blockchain smart contract technology based on resistance to quantum computing attacks. PLoS One. 2024 May 23;19(5):e0302325. doi: 10.1371/journal.pone.0302325. PMID: 38781187; PMCID: PMC11115269.

[12] Shah H, Shah D, Jadav NK, Gupta R, Tanwar S, Alfarraj O, Tolba A, Raboaca MS, Marina V. Deep Learning-Based Malicious Smart Contract and Intrusion Detection System for IoT Environment. *Mathematics*. 2023; 11(2):418. https://doi.org/10.3390/math11020418

[13] Kalpana, P., Anandan, R., Hussien, A.G. *et al.* Plant disease recognition using residual convolutional enlightened Swin transformer networks. *Sci Rep* 14, 8660 (2024). https://doi.org/10.1038/s41598-024-56393-8

[14] Cheng, H., Hu, Q., Zhang, X., Yu, Z., Yang, Y., & Xiong, N. (2021). Trusted Resource Allocation Based on Smart Contracts for Blockchain-Enabled Internet of Things. IEEE Internet of Things Journal. https://doi.org/10.1109/JIOT.2021.3114438

[15] Gopali, S., Khan, Z. A., Chhetri, B., Karki, B., & Namin, A. S. (2022). Vulnerability Detection in Smart Contracts Using Deep Learning. IEEE 46th Annual Computers, Software, and Applications Conference (COMPSAC), 1249-1255. https://doi.org/10.1109/COMPSAC54236.2022.00197

[16] Hasan, Q. O. M. (2023). Machine Learning Based Framework for Smart Contract Vulnerability Detection in Ethereum Blockchain [Thesis]. Rochester Institute of Technology. https://repository.rit.edu/theses/11469

[17] Li, Z., Vott, S., & Krishnamachar, B. (2024). ML2SC: Deploying Machine Learning Models as Smart Contracts on the Blockchain. arXiv. https://doi.org/10.48550/arXiv.2404.16967

[18] Osei, S. B., Ma, Z., & Huang, R. (2024). Smart contract vulnerability detection using wide and deep neural network. Science of Computer Programming, 238, 103172. https://doi.org/10.1016/j.scico.2024.103172

[19] Tang, X., Du, Y., Lai, A., et al. (2023). Deep learning-based solution for smart contract vulnerabilities detection. Scientific Reports, 13, 20106. https://doi.org/10.1038/s41598-023-47219-0

[20] Wu, Z., et al. (2022). Detecting Vulnerabilities in Ethereum Smart Contracts with Deep Learning. 4th International Conference on Data Intelligence and Security (ICDIS), 55-60. https://doi.org/10.1109/ICDIS55630.2022.00016