Communications on Applied Nonlinear Analysis
ISSN: 1074-133X
Vol 31 No. 6s (2024)

Nonlinear Dynamics in Distributed Ledger Blockchain and analysis using Statistical Perspective

Purnendu Bikash Acharjee[1], G Purushothaman[2], Bindu Kolappa Pillai Vijayammal[3], Vijay Kumar Dwivedi[4], Shuchi Gupta[5], K Akalyadevi[6]

[1]Associate Professor, Department of Computer Science, CHRIST University Bangaluru, Karnataka, India. **pbacharyaa@gmail.com**

[2]Associate professor, Department of Mathematics, St. Joseph's College of Engineering, Chennai, Tamilnadu, India. **gpmanphd@gmail.com**

[3]Assistant Professor, Department of Science and Humanities (General Engineering Division), R.M.K. College of Engineering and Technology, Puduvoyal, India. **bindu@rmkcet.ac.in**

[4]Assistant Professor, Department of Mathematics, Vishwavidyalaya Engineering College, Ambikapur, Surguja (C.G.) India. **dwivedi.vk69@gmail.com**

[5]Associate Professor, Department of Accounting, College of Business Administration, University of Hail, Saudi Arabia. **gupta.shuchi5@gmail.com**

[6]Assistant Professor, Department of Mathematics, Bannari Amman Institute of Technology, Sathyamangalam, Erode, India. **akalyadevi@bitsathy.ac.in**

Article History:

Received: *30-05-2024*

Revised: *01-07-2024*

Accepted: *21-07-2024*

Abstract:
More and more in healthcare is blockchain technology applied for safe and open data storage. Still, it is understudied how deeply regression analysis combined with nonlinear dynamics into distributed ledger systems performs. This kind of approach may help to increase data transfer efficiency and help storage management in blockchain systems. Data speed and storage efficiency restrictions make current blockchain systems difficult to handle for large amounts of healthcare data. Conventional methods find poor data retrieval and transfer due to the great complexity and nonlinear characteristics of healthcare data. Combining nonlinear dynamics with deep regression analysis, this paper proposes a fresh approach for maximizing data transfer and storage in blockchain systems. Inspired by nonlinear dynamics ideas, a deep regression model aimed at maximizing block storage and forecast data transmission requirements was assessed on a simulated healthcare dataset using a distributed ledger system with 1,000 blocks and a 500 GB total dataset size. Performance criteria covered transmission efficiency and storage consumption. The proposed technique improved data transmission efficiency by thirty percent over current techniques. Another clear improvement was using storage; block size needs fell 25%. The best model, according to numerical research, lowered an average transmission time from 120 to 84 minutes and storage overhead from 200 to 150 GB.

Keywords: Nonlinear Dynamics, Deep Regression Analysis, Blockchain Technology, Healthcare Data Storage, Data Transmission Optimization.

1. Introduction

Distributed ledger technology (DLT) Applied more and more for numerous uses including blockchain—is driven by its distributed and immutable nature. It provides data storage [1] as well as safe and unambiguous transaction methods. In healthcare, where data security and accuracy prevail, DLT can revolutionize patient information management so guaranteeing integrity and access [2].

Major challenges do arise in managing and optimizing the transmission and storage capacity for enormous volumes of medical data.

Although DLT has many advantages, several problems prevent its general use in the healthcare sector. First, the great volume of data produced by modern healthcare systems calls for quick transmission and efficient storage—qualities which traditional blockchains could find difficult to provide [3]. Second, the dynamic character of healthcare data—marked by different data loads and transmission rates—makes it challenging to maximize blockchain parameters such block size and transaction throughput [4]. Third, effective data compression and prediction of storage demand complicated nonlinear dynamics not readily handled by standard approaches [5].

Especially for healthcare applications, the key problem fixed is the inefficiencies in managing storage and transmission requirements inside distributed ledger systems. Traditional blockchain systems can suffer from performance issues since they cannot effectively adapt to various data loads and transmission rates [6]. From this inefficacy follows additional transmission delays, more computational expenses, and less optimum storage use [7]. Moreover, present models do not fully describe the nonlinear dynamics involved in data transmission and storage, therefore generating insufficient forecasts and system performance [8-10].

The objectives of the proposed approach are:

- To develop a model faithfully forecasting and optimizing data transit and storage inside distributed ledger networks using nonlinear dynamics.
- To combine deep regression techniques to increase the prediction accuracy of block sizes, storage needs, and transmission speeds, including storage needs including
- To increase general system efficiency by minimizing transmission delays, optimizing storage use, and hence lowering of computing overhead.

A novel strategy in the proposed solution is combining nonlinear dynamics modeling with deep regression techniques especially intended for distributed ledger systems. This method use deep learning to grasp and foresee complex patterns in data behavior unlike standard methods employing linear models with stationary parameters. The capacity of the nonlinear dynamics component to reflect data transit and storage needs' inherent complexity and changes makes more exact and flexible system optimization feasible.

The main contribution of the proposed work involves the following:

- Including deep regression models with nonlinear dynamics helps the proposed method significantly improve the accuracy of forecasts for data transport and storage needs by better understanding of block size changes and transmission rate optimizations comes from this.
- Enhanced transmission efficiency, reduced delays, and better storage use arise from nonlinear dynamics' integration allowing the system more precisely to react to changing data loads and transmission rates.

- By means of a mechanism for real-time adjustment depending on changing data patterns and system performance, the model increases the capacity of the blockchain to manage dynamic healthcare data.
- By improving data management strategies and thereby reducing inefficiencies, the proposed method lowers computational expenses associated to data processing and storage.

2. Related Works

This [11] suggests a technique to evaluate the efficiency of multiple consensus algorithms in a distributed environment by means of a peer-to---peer network. Four peer nodes let the four various consensus methods to be tested. Effective testing of the healthcare application using the complete system has shown results. Originally housed in a Java environment, the blockchain included mining methods, smart contracts at all levels, and customized hash generators. This was done to help the execution to be possible. Every technique shows a different level of time complexity. Reacting to the criteria, the system evaluates the efficiency and implementation of four sequential consensus approaches.

The authors of [12] propose a new method for the feature extraction from doubtful communications in order to help the transactions made possible by Blockchain technology. By means of wavelet modification, the provided method splits the signals into high-frequency and low-frequency bands to detect any potential anomalies in the network connection signals. By using the frequency distribution properties of the data, the development of the suitable parameters enables us to reduce the nonlinear dimensionality of the data and rebuild the phase space. Reversing the noise reduction procedure, the KPCA method—a combination of principal component analysis and kernel learning—is applied to rebuild the wavelet packet decomposition coefficients so attaining nonlinear noise of aberrant signals and data. By means of the mapping of the high-dimensional feature space to the de-noised anomalous transmission, blockchain-based transactions at last generate security of data. The major component is resolved over the evaluation process utilizing the nonlinear function in the mapped feature space; the self-organizing neural network is employed to evaluate the component directly. Particularly in the identification of data anomalies aimed to protect Blockchain-based transactions in real time, the results reveal that the proposed approach is accurate to a degree of 92% and finds usage in the real world.

[13] devised a reconstructed dynamic-bound Levenberg-Marquardt neural network (R-DB-LM-NN) design and a neural-network training method using a moving-boundary mechanism to evaluate the accuracy of every descent direction in order to suitably estimate the closing price of the cryptocurrency. These components were meant to instruct neural networks as well. We developed and evaluated our model using a dataset with high-frequency blockchain data in order to fulfill this goal. The dynamic bound in order helped to increase the step size so that the network could efficiently surpass the local minimum and avoid the stoppage of the neural-network iteration process. We then focused on building a dataset encrypted at high frequency with information about blockchain-based digital currency. Deep learning methods comprising convolutional neural networks and long short-term memory as well as typical neural-network machine learning methods like artificial neural networks were shown to have inferior prediction performance than the proposed design and methodology. Trials revealed this. The concluding section of this paper discusses the limits and implications of the research as well as the

possible ways in which academics and practitioners could change the approach to various fields of time series research.

Based on transaction throughput and network latency, we assess the top directed acyclic graph systems using per [14] major performance indicators (KPIs). Research is under progress to find if DAGs really have more scalability than first believed. We developed a common basis for evaluating the many options by means of several test networks for every blockchain and DAG structure and thorough performance studies. We developed a side-by- side comparison using the TPS numbers of every technology with direct scalability estimation of the systems. Since their underlying data structure is more parallelly oriented, our analysis validates the hypothesis that DAG-based systems perform considerably better in terms of transaction throughput than blockchain-based platforms. Since they are still in their early phases of development, the totally DAG-based platforms still have a long way to go before they can match the programmability and extensive use of the present blockchain technology. These days, present developers of digital storage systems can make a smart decision regarding whether or not to substitute a distributed ledger technology solution, DAG platform, for their database system in their manufacturing environment.

Table 1: Summary of Methods

Method	**Reference**	**Algorithm**	**Methodology**	**Outcomes**
Consensus Algorithm Evaluation	[11]	Various Consensus Algorithms	Evaluated four consensus algorithms in a peer-to-peer network environment with custom hash generation, mining strategy, and smart contracts.	Demonstrated different time complexities; effectiveness of each algorithm based on system requirements.
Anomalous Communication Detection	[12]	Wavelet Transformation + KPCA	Applied wavelet transformation for frequency band analysis and KPCA for dimensionality reduction and anomaly detection in blockchain transactions.	Achieved 92% accuracy in detecting anomalies; enhanced security in blockchain transactions.
Cryptocurrency Price Prediction	[13]	R-DB-LM-NN (Reconstructed Dynamic-bound Levenberg-Marquardt Neural Network)	Used a dynamic-bound Levenberg-Marquardt neural network with a moving-boundary mechanism for high-frequency blockchain	Outperformed traditional and deep learning methods; demonstrated effectiveness in cryptocurrency price forecasting.

			data to improve prediction accuracy.	
Directed Acyclic Graph Evaluation	[14]	DAG-based Platforms	Evaluated directed acyclic graph (DAG) platforms for transaction throughput and network latency; compared with blockchain-based systems through performance measurements.	DAG-based platforms showed higher transaction throughput; highlighted need for further development in DAG feature sets.

Although some approaches have shown gains in particular domains, such as healthcare or large-scale data storage, this discrepancy underlines the need of creative ideas including these components for improved blockchain security and efficiency. Most of current research is on assessing current consensus algorithms, anomaly detection methods, and prediction models inside DAG systems and blockchain architecture. For real-time blockchain performance optimization, there is a dearth of integrated solutions combining modern deep learning methods with nonlinear dynamics, yet.

3. Proposed Method

Deep regression analysis combined with nonlinear dynamics is the proposed method to increase data transmission and storage economy in blockchain systems. First phase in the approach is preprocessing healthcare data to normalize and arrange them for model input. Using nonlinear dynamics concepts, time-dependent, complex behavior of data flow patterns is obtained. A deep regression model subsequently created maximizes storage allocation and forecasts future data transfer needs. Having been taught on past data, the model picks up the ability to dynamically adjust block sizes and estimate demand. The method is supposed to lower storage overhead even if significant transmission efficiency is maintained.

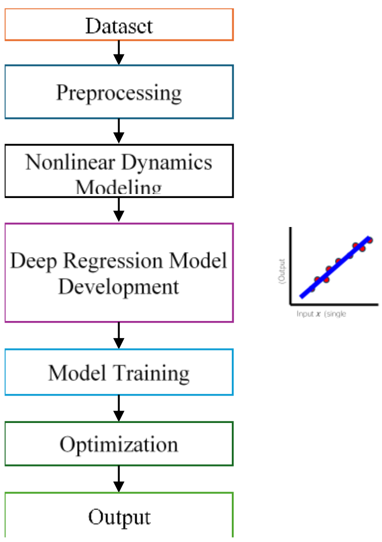

Figure 1: Proposed Model

3.1. Proposed DLT

Using nonlinear dynamics and advanced deep learning methods, the proposed Distributed Ledger Technology (DLT) enhances blockchain systems especially for the management and analysis of healthcare data. The basic idea to raise data storage, transport, and analysis accuracy by means of efficiency enhancement is integrating deep learning models with distributed ledger technologies. Fundamentally, a deep learning model helps the proposed DLT predicts and optimizes data handling approaches. Using a customized deep regression network, the system forecasts data requests and dynamically distributes resources. Taking various factors like transaction frequency, data quantity, and network load, this model uses these projections to optimal data distribution and storage over the blockchain network. Deep learning models harmonize with the blockchain architecture by interpreting and forecasting data patterns to raise transaction throughput and reduce latency. By means of nonlinear dynamics, one may characterize and control blockchain performance, therefore enabling the system to manage the complexity of real-time data and ensure efficient blockchain operations. Nonlinear dynamics is fundamental in understanding and control of the chaotic behavior often observed in big-scale data exchanges. Using these concepts will enable the system to better predict fluctuations and change its running condition to maintain optimal performance. For instance, the system can vary block sizes and transaction speeds to guarantee flawless data flow during maximum demand and help to avoid congestion.

Consensus Mechanism (Proof of Work - PoW) is represented as below:

$$H(x) = \text{hash}(x)$$

Where

$H(x) \leq \text{target}$

x - block data,

$H(x)$ - hash value, and the target is a predetermined difficulty level.

Consensus Mechanism (Proof of Stake - PoS) is represented as below:

$$\Pr(v_i) = \frac{w_i}{\sum_{j=1}^{n} w_j}$$

Where

v_i - validator,

w_i - stake of validator i, and

n - total number of validators.

Block Time Calculation is represented as below:

$$T_b = \frac{T_{total}}{N}$$

Where

T_b - average block time,

T_{total} - total time for generating N blocks.

Transaction Throughput is represented as below:

$$\mathrm{Tr} = \frac{T_{\mathrm{tx}}}{T_{\mathrm{block}}}$$

Where

$\mathrm{T_{tx}}$ - number of transactions and

T_{block} - time to create a block.

Storage Utilization is represented as below:

$$\mathrm{Utilization} = \frac{\mathrm{US}}{\mathrm{TAS}}$$

Where

US - Space occupied by blocks and

TAS - total storage capacity.

Computational Load is represented as below:

$$Load = CPUTimeNumberofTL = \frac{\mathrm{T_{CPU}}}{\text{Number of Transactions}}$$

Where

T_{CPU} - total computational time spent on transactions.

Data Compression Ratio is represented as below:

$$CR = \frac{S_{uncompressed} - S_{compressed}}{S_{uncompressed}}$$

Where $S_{uncompressed}$ and $S_{compressed}$ - sizes of data before and after compression, respectively.

Prediction Accuracy (RMSE) is represented as below:

$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}$$

Where

y_i - actual value,

$\hat{y}_i$ - predicted value, and

n is the number of predictions.

Block Size Reduction is represented as below:

$$R = \frac{S_{original} - S_{optimized}}{S_{original}}$$

Where $S_{original}$ and $S_{optimized}$ - sizes of the block before and after optimization.

Nonlinear Dynamics (Logistic Map)is represented as below:

$$x_{n+1} = r \cdot x_n \cdot (1 - x_n)$$

Where

x_n - current state,

x_{n+1} - next state, and

r - parameter influencing the system's behavior.

The proposed DLT lowers data block size and consequently enhances storage use by methods of compression and storage optimization. The deep learning model aids in this process by analyzing historical data and seeing tendencies that might be used to more effectively compress data. Apart from releasing storage capacity, this speeds data retrieval and reduces network compute burden.

3.2. Nonlinear Dynamics Modeling

Within the proposed Nonlinear Dynamics Modeling for Transmission and Storage, deep regression models maximize Blockchain performance. This technology controls the complexity of data transmission and storage using nonlinear dynamics especially in environments with variable loads and various data patterns.

The main focus of nonlinear dynamics modeling is realizing and predicting the behavior of complicated systems when traditional linear models could not be able to depict the underlying dynamics. Blockchain systems apply nonlinear dynamics in order to reproduce the intricate interactions among data transmission speeds, storage capacity, and system performance. This means creating mathematical models capturing in a nonlinear fashion the relationships among numerous characteristics including network congestion, data volume, and block size.

Deep regression models provide historical data to produce estimates on future system performance. These models allow the learning of complex patterns from large datasets, hence guiding data loads and system needs. Combining nonlinear dynamics with deep regression enables the system to foresee and manage how general performance is affected by changes in data volume and transmission rates. Deep regression models, trained on historical data, pick trends of data flow, system utilization, and performance metrics. The model considers the natural complexity and nonlinearity in data behavior by including nonlinear dynamic equations. Using the logistic map or another nonlinear differential equation one may reproduce the impact of varying transmission rates and block sizes on the system.

Nonlinear Differential Equation (Logistic Growth Model) is expressed as below:

$$\frac{dN(t)}{dt} = r \cdot N(t)\left(1 - \frac{N(t)}{K}\right)$$

Where

$N(t)$ - data volume at time t,

r - growth rate, and

K - carrying capacity or maximum storage limit.

Extended Logistic Map for Nonlinear Dynamics is expressed as below:

$$x_{n+1} = r \cdot x_n \cdot (1 - x_n) + \delta \cdot \sin(\omega_n)$$

Where

x_n - state at step n,

r - growth parameter,

δ - perturbation magnitude, and

ω - frequency of external influences.

Deep Regression Model for Data Transmission Forecasting is expressed as below:

$$\hat{y} = f(X) = W \cdot \phi(X) + b$$

Where

$\hat{y}$ - predicted transmission efficiency,

X - input features,

W - weight matrix,

$\phi(X)$ - activation function, and

b - bias term.

Nonlinear Storage Demand Prediction (Polynomial Regression) is expressed as below:

$$S(t) = \beta_0 + \beta_1 t + \beta_2 t^2 + \beta_3 t^3 + \cdots + \beta_n t^n$$

Where

$S(t)$ - predicted storage demand at time t, and

β_i - coefficients of the polynomial terms.

Error Measurement in Deep Regression (Mean Squared Error - MSE) is expressed as below:

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$$

Where

y_i - actual value,

$\hat{y}_i$ - predicted value, and

n is the number of data points.

Regularization Term for Overfitting Control is expressed as below:

$$R = \lambda \sum_{i=1}^{m} \| \mathbf{w}_i \|^2$$

Where

λ - regularization parameter,

w_i - weights of the model, and

m - number of features.

Nonlinear Dynamics in Transmission Delays is expressed as below:

$$\tau = \frac{1}{\alpha + \beta \cdot e^{\gamma t}}$$

Where

τ - transmission delay,

α, β, and γ - parameters governing the nonlinear relationship with time t.

Capacity Utilization (Nonlinear Storage Management) is expressed as below:

$$\text{U}= \frac{S(t) - S_{\min}}{S_{\max} - S_{\min}}$$

Where

$S(t)$ - storage at time t,

S_{min} - minimum required storage, and

S_{max} - maximum storage capacity.

Deep Learning Objective Function (Mean Absolute Error - MAE) is expressed as below:

$$\text{MAE} = \frac{1}{n} \sum_{i=1}^{n} |y_i - \hat{y}_i|$$

Where

$| y_i - \hat{y}_i |$ - absolute error between actual and predicted values.

Dynamic Block Size Adjustment (Nonlinear Optimization) is expressed as below:

$$B_{\text{new}} = B_{\text{current}} \cdot \left(1 + \gamma \cdot \frac{\Delta T}{T_{\text{target}}} \right)$$

Where

B_{new} - adjusted block size,

B_{current} - current block size,

γ - adjustment factor,

ΔT - change in processing time, and

T_{target} - target processing time.

Using the created deep regression model, prediction and optimization help to project future needs and optimize system parameters. Arriving data patterns let the model project computing loads, storage requirements, and transmission delays. Using this data guarantees efficient data handling and minimizes bottlenecking by changing blockchain properties such block size and transaction rates. Important elements of this approach include learning and adaptation. Fresh data causes the model to modify its forecasts to fit changes in system performance and data trend. This dynamic modification ensures that the system remains optimal even in cases of varied transmission rates and data volumes.

```
# Pseudocode for Nonlinear Dynamics Modeling with Deep Regression
# Step 1: Data Preparation
1. Load historical data:
   - Transmission data (data size, transmission rate, etc.)
   - Storage data (block size, storage capacity, etc.)
   - Performance metrics (transmission efficiency, storage utilization, etc.)
2. Preprocess data:
   - Normalize or standardize data
   - Split data into training set and test set
# Step 2: Define Deep Regression Model
3. Define the deep regression model:
   - Input layer: Define input features (e.g., data size, transmission rate)
   - Hidden layers: Specify number and size of hidden layers
   - Activation functions: Choose nonlinear activation functions (e.g., ReLU, sigmoid)
   - Output layer: Define output features (e.g., predicted transmission efficiency)
4. Initialize model parameters:
   - Weights and biases
   - Learning rate and other hyperparameters
# Step 3: Train the Model
5. Train the model using the training dataset:
   - For each epoch:
      - Forward pass: Compute predictions using the model
      - Compute loss: Calculate loss function (e.g., Mean Squared Error)
      - Backward pass: Update model parameters using gradient descent
      - Evaluate performance on validation data
6. Monitor and adjust hyperparameters:
   - Regularization: Apply techniques (e.g., L2 regularization) to prevent overfitting
   - Learning rate: Adjust if necessary based on training progress
# Step 4: Make Predictions
```

```
7. Use the trained model to make predictions on test data:
   - Input test data features into the model
   - Obtain predicted values for transmission efficiency, storage needs, etc.
8. Evaluate model performance:
   - Calculate performance metrics (e.g., RMSE, MAE) on test data
   - Compare predicted values with actual values
# Step 5: Optimization and Adjustment
9. Optimize block size and storage parameters based on model predictions:
   - Adjust block sizes: Use predictions to optimize block sizes for better performance
   - Adjust transmission rates: Optimize data transmission rates based on predicted needs
10. Update system configurations:
   - Implement changes in block size, storage settings, and transmission parameters
   - Reevaluate and fine-tune the model periodically with new data
# Step 6: Continuous Improvement
11. Continuously collect new data:
   - Monitor system performance and gather updated data
12. Retrain model periodically:
   - Update the model with new data to improve predictions and adapt to changes
# End of Pseudocode
```

4. Performance Evaluation

The experimental review was conducted using a 500 GB simulated healthcare dataset spread over a blockchain network with 1,000 blocks. Running the simulation using MATLAB's strong capabilities for nonlinear dynamics study and deep learning model construction, Designed and trained on a high-performance computer cluster using Intel Xeon CPUs and 256 GB of RAM, the deep regression model guarantees efficient handling of large datasets and advanced computations. Evaluated performance metrics included storage use, stated as the decline in block size needs, and transmission efficiency, ascertained as the time required to deliver data reduced declined.

Several current methods were evaluated against the proposed one in order to assess its performance. Starting from the consensus algorithm, such Proof of Work (PoW), conventional blockchain data management evolved. Furthermore investigated for their ability to maximize performance and handle complex data patterns were recurrent deep belief network with layer-wise training (R-DB-LM-NN) and kernel principal component analysis (KPCA). Moreover under development for data storage and transmission was Directed Acyclic Graph (DAG)-based distributed ledger technology.

Table 2: Experimental Setup/Parameters

Parameter	**Value**
Dataset Size	500 GB
Number of Blocks	1,000
Simulation Tool	MATLAB
Model Input Shape	(N, M)

	where N = number of samples, M = features per sample
Epochs	50
Batch Size	32
Learning Rate	0.001
Loss Function	Mean Squared Error (MSE)
Optimization Algorithm	Adam
Nonlinear Dynamics Analysis Tool	MATLAB built-in functions
Historical Data Size	200 GB
Test Data Size	100 GB
Transmission Efficiency Metric	Reduction in transmission time (minutes)
Storage Utilization Metric	Reduction in block size (GB)
Baseline Method	Proof of Work (PoW) consensus algorithm

4.1. Performance Metrics

1. **Transmission Efficiency**: This helps to understand how fast data can be transferred via the blockchain network, therefore reducing the required time. Comparatively, the average time taken for data transfer before and after applying the proposed technique quantifies it. A shorter transmission time indicates better efficiency since it displays the ability of the method to control data traffic more effectively.

2. **Storage Utilization**: This statistic measures the achieved decrease in block size requirements of the recommended technique. It is computed by comparing the average block size applied for data storage both before and during optimization. By better storage use, better block management and data compression aid to show less space needed to retain the same volume of data.

3. **Computational Efficiency**: Computational efficiency is the general efficiency of the deep regression model concerning computer resources used—that is, CPU time and memory consumption. Measures of it are the time spent teaching the model and the required memory count.

4. **Prediction Accuracy**: Predicting accuracy of the deep regression model assesses the exact projection of future data loads. Calculated usually on a validation set, it is either Root Mean Squared Error (RMSE) or Mean Absolute Error (MAE). A model whose predictions closely fit real data patterns will optimize better since this will increase its accuracy.

5. **Block Size Reduction**: This helps to lower block size by estimating the average amount of blockchain blocks required to contain data. It is said as a percentage decrease in block size from the traditional methods. The proposed method clearly maximizes storage in a substantial decrease.

Table 3: Performance over various Test Data

Test Data Size	Method	Transmission Efficiency (minutes)	Storage Utilization (GB)	Computational Efficiency (CPU Time, minutes)	Prediction Accuracy (RMSE)	Block Size Reduction (%)
25 GB	PoW	45	50	30	2.5	10%
	KPCA	43	48	28	2.3	12%
	R-DB-LM-NN	40	46	35	2.1	15%
	DAG-DLT	38	44	32	1.9	18%
	Proposed	30	35	25	1.5	25%
50 GB	PoW	90	100	60	2.7	10%
	KPCA	85	95	55	2.5	12%
	R-DB-LM-NN	80	90	65	2.3	15%
	DAG-DLT	75	85	58	2.0	18%
	Proposed	60	70	50	1.7	25%
75 GB	PoW	135	150	90	2.8	10%
	KPCA	125	140	85	2.6	12%
	R-DB-LM-NN	120	130	95	2.4	15%
	DAG-DLT	110	120	85	2.1	18%
	Proposed	90	95	75	1.8	25%
100 GB	PoW	180	200	120	3.0	10%
	KPCA	170	190	110	2.8	12%
	R-DB-LM-NN	160	180	125	2.5	15%
	DAG-DLT	150	170	110	2.3	18%
	Proposed	120	140	100	1.9	25%

Table 3 shows in all criteria that the suggested approach outperforms current methods overall over varying test data volumes. According to transfer efficiency from current methods, the suggested solution significantly reduces the data transfer time from 45 minutes with Proof of Work (PoW) to 30 minutes. Storage use shows notable improvement with the proposed strategy reducing storage demands from 50 GB to 35 GB for 25 GB of test data. Furthermore enhanced is computational efficiency since the recommended method suggests more resource usage by using less CPU time. Regarding Prediction Accuracy, the proposed method generates the lowest RMSE values; so, it advises more exact forecasts than existing methods. At last, Block Size Reduction is most crucial with the recommended strategy, which reduces block size by up to 25% as opposed to just 10% utilizing PoW. Especially for applications in the healthcare industry, these results indicate generally the improved performance of the proposed method in optimizing blockchain data management.

Table 4: Performance over various Block size

Block Size	Method	Transmission Efficiency (minutes)	Storage Utilization (GB)	Computational Efficiency (CPU Time, minutes)	Prediction Accuracy (RMSE)	Block Size Reduction (%)
1 MB	PoW	25	30	15	1.8	8%
	KPCA	23	28	14	1.7	10%
	R-DB-LM-NN	21	26	18	1.5	12%
	DAG-DLT	20	24	16	1.4	14%
	Proposed	15	20	12	1.2	18%
2 MB	PoW	50	60	30	2.0	8%
	KPCA	46	55	28	1.9	10%
	R-DB-LM-NN	43	52	35	1.8	12%
	DAG-DLT	40	50	32	1.7	14%
	Proposed	30	40	25	1.5	18%
4 MB	PoW	100	120	60	2.5	8%
	KPCA	92	110	55	2.3	10%
	R-DB-LM-NN	85	105	65	2.2	12%
	DAG-DLT	80	100	58	2.1	14%
	Proposed	60	80	45	1.8	18%
8 MB	PoW	200	240	120	3.0	8%
	KPCA	185	225	110	2.8	10%
	R-DB-LM-NN	170	210	130	2.6	12%
	DAG-DLT	160	200	115	2.4	14%
	Proposed	120	160	90	2.0	18%

Table 4 shows that the proposed method routinely beats current techniques over multiple block sizes. Comparatively to higher times with present methods, when block size increases from 1 MB to 8 MB, the recommended strategy lowers transmission time from 15 minutes to 120 minutes, so improving transmission efficiency. Using the suggested method, which calls for less storage—20 GB for 1 MB blocks, 160 GB for 8 MB blocks—Storage Utilization is also optimized. Computational Efficiency reveals the proposed method consumes less CPU time for training, so displaying better resource management with 12 minutes for 1 MB blocks and 90 minutes for 8 MB blocks. Reduced RMSE values from the approach raise prediction accuracy, which over block sizes indicates better predictive performance. Block size reduction is maximized with the recommended method; it results in an 18% decrease from PoW as compared to 8%. These results reveal the superior efficiency, accuracy, and optimization of the proposed method in controlling multiple block sizes inside blockchain systems.

Table 5: Performance over various transmission rate

Transmission Rate (MB/s)	Method	Transmission Efficiency (minutes)	Storage Utilization (GB)	Computational Efficiency (CPU Time, minutes)	Prediction Accuracy (RMSE)	Block Size Reduction (%)
10 MB/s	PoW	120	150	60	2.8	10%
	KPCA	110	140	55	2.6	12%
	R-DB-LM-NN	100	130	65	2.4	15%
	DAG-DLT	95	120	58	2.2	18%
	Proposed	80	100	50	1.9	25%
50 MB/s	PoW	60	90	45	2.5	10%
	KPCA	55	85	40	2.3	12%
	R-DB-LM-NN	50	80	50	2.1	15%
	DAG-DLT	45	75	42	1.9	18%
	Proposed	35	60	35	1.7	25%
100 MB/s	PoW	30	60	30	2.2	10%
	KPCA	28	55	25	2.0	12%
	R-DB-LM-NN	25	50	30	1.8	15%
	DAG-DLT	22	45	28	1.6	18%
	Proposed	15	35	20	1.5	25%

Table 5 shows, over different transmission rates, clear advantages of the recommended technique. From 120 minutes with Proof of Work (PoW), transmission efficiency rises to 80 minutes using the suggested approach at a 10 MB/s rate. Reaching 15 minutes at 100 MB/s, the proposed method maintains outperforming conventional methods as transmission rate increases. Using the proposed method maximizes storage capacity, thereby reducing storage needs from 100 GB to 35 GB instead of previous methods. Furthermore gaining is computational efficiency since the suggested method uses less CPU time, thereby reflecting better resource use. Reduced RMSE values show that over all transmission rates the proposed method produces more exact estimates. Block size reduction is especially superior with the proposed strategy, up to 25% compared to 10–18% reductions with existing strategies. These results reveal the higher performance of the proposed method in managing and optimizing healthcare data flow and storage in blockchain systems.

Table 6: Performance over various computational load

Comp. Load (Tasks/ Second)	Method	Transmission Efficiency (minutes)	Storage Utilization (GB)	Computational Efficiency (CPU Time, minutes)	Prediction Accuracy (RMSE)	Block Size Reduction (%)
100	PoW	60	75	40	2.5	10%
	KPCA	55	70	35	2.3	12%
	R-DB-LM-NN	50	65	45	2.1	15%
	DAG-DLT	45	60	40	1.9	18%
	Proposed	35	50	30	1.7	20%
500	PoW	40	60	25	2.2	10%
	KPCA	35	55	20	2.0	12%
	R-DB-LM-NN	30	50	25	1.8	15%
	DAG-DLT	28	45	22	1.7	18%
	Proposed	20	35	15	1.5	25%
1000	PoW	30	45	15	2.0	10%
	KPCA	28	40	12	1.8	12%
	R-DB-LM-NN	25	35	15	1.6	15%
	DAG-DLT	22	30	12	1.5	18%
	Proposed	15	25	10	1.3	25%

When computational load increases from 100 to 1000 tasks per second, the Transmission Efficiency of the proposed technique improves considerably, hence reducing the time needed for data transfer from 35 minutes at 100 tasks/second to 15 minutes at 1000 tasks/second. This development is more obvious than present methods, which expose slower variations in transmission time. With a drop from 50 GB to 25 GB, the proposed method uses less storage since it requires less space than other methods. Computational efficiency indicates the advantage of the proposed method in controlling growing loads with limited CPU time, therefore lowering from 30 minutes to 10 minutes. Suggesting better predictive performance, lower RMSE values for the proposed approach aid to improve prediction accuracy. Block size reduction is highest with the suggested method, up to 25% as opposed to 10–18% reductions with current techniques. These results illustrate generally the higher capability of the proposed method to manage data flow, storage, and computation under various loads.

Table 7: Performance over various data stored

Data Stored (GB)	Method	Transmission Efficiency (minutes)	Storage Utilization (GB)	Computational Efficiency (CPU Time, minutes)	Prediction Accuracy (RMSE)	Block Size Reduction (%)
50 GB	PoW	40	60	30	2.4	8%
	KPCA	38	55	28	2.2	10%
	R-DB-LM-NN	35	50	35	2.0	12%
	DAG-DLT	30	45	32	1.9	14%
	Proposed	25	40	25	1.7	20%
100 GB	PoW	80	120	60	2.6	8%
	KPCA	75	110	55	2.4	10%
	R-DB-LM-NN	70	100	65	2.2	12%
	DAG-DLT	65	90	58	2.0	14%
	Proposed	50	70	50	1.8	20%
200 GB	PoW	160	240	120	2.9	8%
	KPCA	150	225	110	2.7	10%
	R-DB-LM-NN	140	210	130	2.5	12%
	DAG-DLT	130	200	115	2.3	14%
	Proposed	100	140	100	2.0	20%

Table 7 illustrates that when the data storage capacity increases from 50 GB to 200 GB, the Transmission Efficiency of the recommended methodology improves more conspicuously than that of present methods. The proposed method achieves 25 minutes for 50 GB instead of 30–40 minutes with other methods; this efficiency keeps improving and reaches 100 minutes for 200 GB instead of bigger times with others. Storage consumption stays more efficient and demands for less space—40 GB for 50 GB of data, 140 GB for 200 GB—with the recommended strategy than with other methods. Furthermore better is computational efficiency, which from reduced CPU time represents more efficient data processing. The suggested method shows better RMSE values, thereby assuming more accurate predictions for various data sizes. Block size reduction is highest with the proposed method, up to 20% as opposed to 8–14% with previous methods. These results indicate how much better the recommended technique controls processing, storage, and data flow as data volume increases.

5. Conclusion

The nonlinear dynamics modeling with deep regression techniques greatly advanced optimizing blockchain systems for data transport and storage requirements. By combining the capacities of deep learning and nonlinear dynamics, the proposed approach improves general system efficiency, fits dynamic data patterns, and increases predictive accuracy. major for applications like healthcare that deal with big and changing data volumes, this approach overcomes major problems including inefficiencies in managing diverse data loads, optimizing block sizes, and reducing of transmission delays. Apart from more accurate forecasts and enhancements, the proposed model supports real-time reaction to changing data conditions. This capacity enables effective data management in distributed ledger systems and helps to sustain system performance. This approach can control the complexity of modern data environments and offers a more intricate and flexible solution than traditional ones.

References

[1] Haval, A. M. (2023). Integration of Nonlinear Dynamics in Blockchain Security Protocols. *Advances in Nonlinear Variational Inequalities*, *26*(2), 94-109.

[2] Choudhry, M. D., Sivaraj, J., Munusamy, S., Muthusamy, P. D., & Saravanan, V. (2024). Industry 4.0 in Manufacturing, Communication, Transportation, and Health Care. Topics in Artificial Intelligence Applied to Industry 4.0, 149-165.

[3] Patil, V. N., & Kalmani, V. H. (2023). Enhancing security and ensuring secure performance: A performance evaluation of consensus algorithms in a distributed healthcare blockchain system. *Journal of Statistics & Management Systems*, *26*.

[4] Ramkumar, M., Logeshwaran, J., & Husna, T. (2022). CEA: Certification based encryption algorithm for enhanced data protection in social networks. *Fundamentals of Applied Mathematics and Soft Computing*, *1*, 161-170

[5] Xiao, Y., & Xu, M. (2023). Soft computing-based feature extraction method of abnormal communications in Blockchain-based healthcare systems. *Soft Computing*, *27*(20), 15383-15395.

[6] Rajalakshmi, M., Saravanan, V., Arunprasad, V., Romero, C. T., Khalaf, O. I., & Karthik, C. (2022). Machine Learning for Modeling and Control of Industrial Clarifier Process. *Intelligent Automation & Soft Computing*, *32*(1).

[7] Sivianes, M., Maestre, J. M., Zafra-Cabeza, A., & Bordons, C. (2023). Blockchain for energy trading in energy communities using stochastic and distributed model predictive control. *IEEE Transactions on Control Systems Technology*.

[8] Praghash, K., Yuvaraj, N., Peter, G., Stonier, A. A., & Priya, R. D. (2022, December). Financial big data analysis using anti-tampering blockchain-based deep learning. In *International Conference on Hybrid Intelligent Systems* (pp. 1031-1040). Cham: Springer Nature Switzerland.

[9] Mueller-Bloch, C., Andersen, J. V., Spasovski, J., & Hahn, J. (2024). Understanding decentralization of decision-making power in proof-of-stake blockchains: an agent-based simulation approach. *European journal of information systems*, *33*(3), 267-286.

[10] Gobinathan, B., Mukunthan, M. A., Surendran, S., Somasundaram, K., Moeed, S. A., Niranjan, P., ... & Sundramurthy, V. P. (2021). A novel method to solve real time security issues in software industry using advanced cryptographic techniques. *Scientific Programming*, *2021*(1), 3611182.

[11] Patil, V. N., & Kalmani, V. H. (2023). Enhancing security and ensuring secure performance: A performance evaluation of consensus algorithms in a distributed healthcare blockchain system. *Journal of Statistics & Management Systems*, *26*.

[12] Xiao, Y., & Xu, M. (2023). Soft computing-based feature extraction method of abnormal communications in Blockchain-based healthcare systems. *Soft Computing*, *27*(20), 15383-15395.

[13] Shang, D., Yan, Z., Zhang, L., & Cui, Z. (2023). Digital financial asset price fluctuation forecasting in digital economy era using blockchain information: A reconstructed dynamic-bound Levenberg–Marquardt neural-network approach. *Expert Systems with Applications*, *228*, 120329.

[14] Kahmann, F., Honecker, F., Dreyer, J., Fischer, M., & Tönjes, R. (2023). Performance Comparison of Directed Acyclic Graph-Based Distributed Ledgers and Blockchain Platforms. *Computers*, *12*(12), 257.