Tuijin Jishu/Journal of Propulsion Technology
ISSN: 1001-4055
Vol. 44 No. 5 (2023)

Leveraging Blockchain Driven Stacked Model For Ransomware Detection In Bitcoin Transactions

[1] N. Sivakumar, [2] Dr. G. Jagatheeshkumar

[1]Research Scholar, PG & Research Department of Computer Science
Karuppannan MAriappan College, Muthur, Tamilnadu, India
[2] Research Guide, Associate Professor & Head,
PG & Research Department of Computer Science, Karuppannan Mariappan College,
Muthur,Tamilnadu, India.

Abstract: To improve interoperability and privacy among system users, smart grids must allow for the sharing of data and information. Traditional cloud-based data interchange techniques, on the other hand, have been centralised on a single platform run by a dependable third party, which has resulted in single points of failure, inadequate data protection, and unrestricted access. Blockchain technology has been suggested as a decentralised and secure platform for data exchange within smart grids to overcome these problems. This innovative platform offers solutions to important issues like privacy, scalability, and user ownership and enables safe data trading between users without ownership loss. Participants can access data programmatically with the help of blockchain-based smart contracts while guaranteeing that every interaction is verified and documented by other users of the tamper-resistant blockchain network. In this regard, a new model for forecasting S&P500 volatility has been put out employing a variety of machine learning methods, including Gradient Descent Boosting, Random Forest, Support Vector Machine, and Artificial Neural Network. In order to increase the predictability of the predictions, these algorithms have been stacked. Resilient K-NN and FLR have been combined in the suggested stacking model for prediction, and it has demonstrated classification accuracy of 97.41% and 88.27%, respectively. These experimental findings suggest that it is advantageous to look into the use of PART for tasks involving predictive modelling in Ransomware investigations.

Keywords: Ransomware, PART, stacking, blockchain, bitcoin network, decentralization and gradient descent.

1. Introduction

Over the years, a variety of digital money models have been suggested and put into use. Using a consensus method based on evidence of work, block chain technology enables the ability of decentralised money to exist, conduct reliable transactions, and avoid double spending [1, 2].Because of its absence of regulation and the involvement of criminals, Bitcoin has recently grabbed investors' attention and scholars.Ransomware is a category of malware that, after successfully infecting The computer system of a target encrypts files and personal information so that the victim could no longer control them. There has been a recent spike in new virus types that target the Internet.The problem is then described in a message that is delivered to the victim, along with instructions on how to regain full control of their own machine, which typically entails making a payment using Bitcoin.There are several hacker families and hacker groups using this software, but their defining trait is the demand for payment in order to access the stolen data [3].

In order to evaluate if it is possible to categorise each transaction as being a part of a ransomware family or not, we propose to analyze a number of transactions that have been obtained from the Bitcoin blockchain.Finding out who submitted a bitcoin payment or when it was made is difficult because the transactions are anonymous.Because of this, it is essential that these transactions be identified and tagged in accordance with if they are trading operations or authorised exchanges in order to be able to recognise and categorise them accurately.In this article, machine learning (ML) algorithms are used to identify that whether transfer is malicious or benign [4, 5].We utilised the dataset from the Bitcoin ransomware to categorise the various forms of harmful Bitcoin transactions.The analysis of various transactional aspects. The results are assessed for precision, accuracy, and recall.

The proposed research works' contributions are as follows:

- A new method has been proposed for sharing Data and information exchanged between different organizations in smart grids, which involves utilizing the Bitcoin network's blockchain technology.
- The system employs data off-chain archiving with the Hadoop Distributed File System and a permissioned blockchain, which allows for improved performance.
- To further enhance the efficiency of the blockchain platform, ANN-based optimisation methodology has been put into practise. The suggested system can connect with external machine-learning components thanks to its modular and extendable architecture.
- Compared to other hybrid models, the Stacked-ANN uses the ANN's direct input of the projections produced by the first-level algorithms. In contrast, most hybrid models involve inserting portions of the GARCH-based model into the ANN independently.

Here is the structure of the remaining text: An examination and identification of ransomware are included in the literature study in Section 2.Proposed system specifications and modelling are covered in Section 3.Section 4 presents the outcomes and the performance analysis.Part 5 provides the research's the end and the future directions.

2. Review of related works

While being often utilised in the literature for a variety of purposes and disciplines, rule-based algorithms have not yet developed to the point where they can be applied to classification issues in the Bitcoin Ransomware arena.In addition, various machine learning algorithms were used to identify and detect Ransomware.This section examines pertinent articles that used Rule-Based models in different disciplines and machine learning approaches for Ransomware detection.For the purpose of identifying recent harmful addresses in the Ransomware family, [6] used topological data analysis approaches.A directed weighted graph was used by the authors to create a model of the Bitcoin graph.Payments made to well-known addresses of the ransomware family are used to locate new addresses of the virus family.The first 20,000 clusters of Ransomware addresses are created.The clusters that come from this analysis are then looked at to determine if any Ransomware families are related.Both Topographic Data Analysis (TDA) as well as the DBSCAN clustering approach (Density-Based Spatially Clustering of Applications using Noise) are used to detect and forecast Ransomware in financial transactions.Their proposed method, which could be used to automatically detect ransomware, greatly increased accuracy and recall for ransomware transaction recognition when compared with current heuristic-based techniques.

The CryptoLocker malware family was analysed by [7] in their study. A programme that recognises the Bitcoin addresses used for CryptoLocker ransom payments automatically.Performance assessments on the data were examined using block chain analysis and information retrieved from websites like Reddit and BitcoinTalk.The timestamps are obtained based on the ransom payments made by the victims.These data were used to pay and analyse historical ransom amount patterns.A measuring analysis of Ransomware payments over a two-year period, including data on victims and operators, was carried out by [8].A large list of Bitcoin addresses, victim telemetry, and Ransomware files were used to construct a sizable dataset.By using this data, the operators were able to follow the victim's Bitcoin transactions from the moment the victim got the currency until it was cashed out.When the two algorithms are compared, the current one produces better and more accurate results than the old one.[9] used unique method, NetConverse, uses a J48-based decision tree classifier to identify Ransomware samples from attributes extracted from network traffic communication.According to their experimental findings, the proposed method recognised more accurately than other well-known algorithms for machine learning such as the Network of Bayes, Neighbor of K-Nearest, Feedforward Neural Network, Random Decision Forest, Logit Regression.

The two cutting-edge Rule-Based approaches for intrusion detection were employed by [10] in their 2019 study.Information gained from the ranker attribute evaluator was used in the feature selection.PART and Decision Table performed effectively in detecting intrusion, according to the experimental data.In addition, the Decision table delivered 99.99% accuracy, outperforming the previous algorithms.To detect fraud in transactions using credit cards, [11] developed a number of machine learning & rule-based models.By selecting the most suitable algorithm to be included in the fraudulent identification system, they attempted to enhance its capacity of detection of fraud using a variety of machine learning techniques.The two well-known rule-based algorithms produced

respectable results in the experimental findings, yielding 81.95% efficiency for Decision Table classifier & 81.37% for the PART classifier.

By using logistic regression prediction assessment analysis and rule-based classifications (Segment & Decision tables) on a data mining platform, [12] were able to identify prospective diabetes as well as pre-diabetes in the primary medical monitoring.Using 10 practical and easily obtained non-invasive clinical features collected from four significant hospitals in northern Nigeria, 281 patients with diabetes mellitus were evaluated.Diabetes Care published the findings there.98.75% accuracy was the maximum that Rule-Based classifiers could achieve, per the experimental findings. Also, the error rate was 0.98%, the precision was 0.98%, the recall was 0.98%, and the F-measure was 98%.

The Bagging Ensemble method was utilised by Gaikwad and Thool (2015) to develop a system for intrusion detection. The Partial Decision Tree was used as the basis for classification because of how straightforward it is.To improve the classifier's accuracy, the pertinent features are chosen using an optimization approach.Classification accuracy, false positives, true positives and model building time are all included in the assessment for the suggested intrusion detection system.When compared to other classifiers using the suggested framework only with Partial Decision Tree (PART) method, the suggested system using the PART strategy achieved the highest level of cross-validation classification accuracy of 99.71%.The Decision Table technique was one of ten different Machine Learning algorithms that [13] investigated.In order to compare the algorithms and choose the best one, the researchers put the method to the test on the MovieLens dataset.After the classification procedure, various evaluation measures, such as the F-measure, Kappa Statistic, and Accuracy, were used.With 98.79% accuracy, Decision Table achieved a respectable performance in the experiment.

Several machine learning methods have presumably been employed for Ransomware categorization tasks, according to earlier studies.Outstanding findings were reached by some of those investigations.Examples of studies with positive results are also included.However, several academics have applied Rule-Based algorithms in different fields and had great success. For this goal, we chose to bridge that gap and implement PART [14, 15] as well as Rule-based classifiers to recognise and identify Ransomware with in selected dataset.The following sections discuss results that are related.

3. Proposed blockchain architecture for bit coin networks

The system architecture that permits data sharing and exchange among smart grid entities is shown in this part. This system combines block chain technology with HDFS for off-chain storage.The Hyper Ledger Fabric (HLF) block chain and Apache Hadoop are the foundation of the new architecture.The exchange of data in the smart grid is outlined in this study. By implementing third-party computing inside the environment of the data owner, the system design guarantees the privacy and the data's protection. Organizations involved in the energy grid, like DSOs and TSOs, can gain from this venue.Blockchain technology Electronicsand smart agreements enable openness regarding the sharing and access to data, allowing consumers to find out who, when, and why its information was viewed.

Data creators: DSOs, TSOs, and other market players are the platform's data creators. The members of the network add to the production of data. This data can either be recovered and shared with other network users or uploaded for storage to a cloud computer and a blockchain network.

Exchange Platform: Various smart grid organisations collect data, which is stored on cloud servers. Using a block chain, the index record of the data is kept up to current and possibly contains details about the position of the data storage. This gives the network a way to store info off-chain.Data consumers: Additionally, the government must identify the data consumers network. Whether the plan is accepted or rejected determines whether the data will be accessible to data consumers.Figure 2 [37] depicts the registration procedure for the system users a participant in the proposed network.

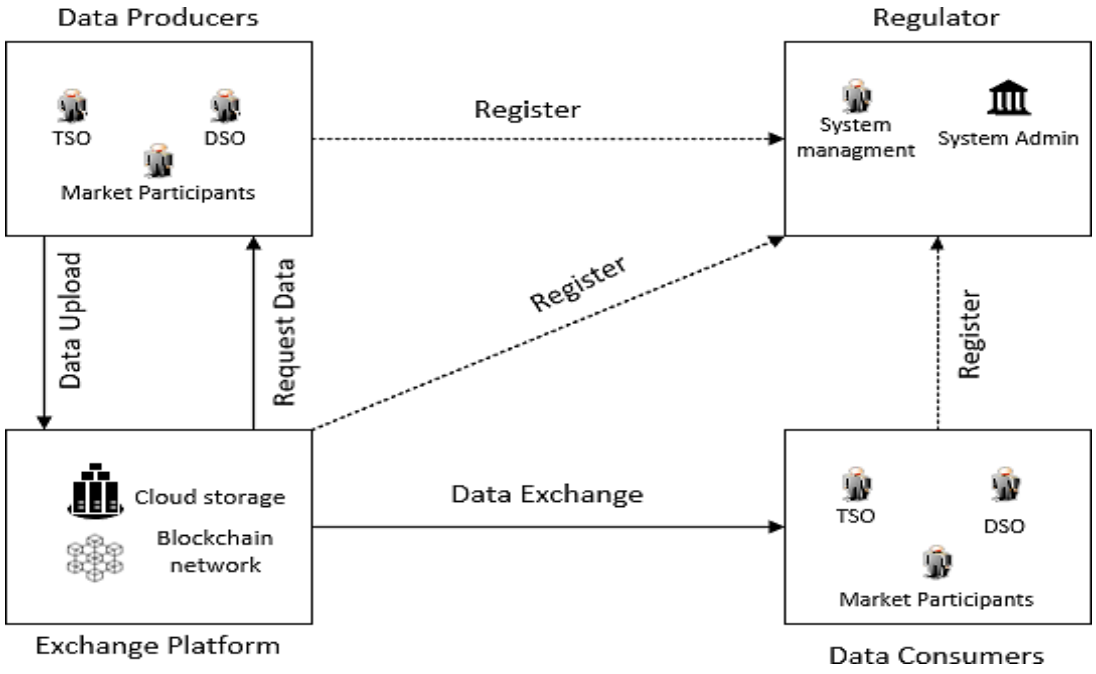

Figure 1: Entities of the system

The architecture for data and information sharing inside smart grids is shown in Figure 1 [38]. The processing, blockchain, and data storing components make up this system. Data providers, consumers, TSOs, DSOs, and market players are some of the main users of the blockchain consortium. The customer enters data, which is then forwarded to Hadoop to analyze the blockchain. The provided data collection may only be used by data consumers for which the system has granted authorizationA document is deployed by the data provider, and it looks at the client's programming.By keeping an eye on possibly hazardous elements in user code and reduces the intricacy of computing. It was a clever contract made using a combination of data sources. The proposed solution includes the HDFS storage layer, responsible for data archiving. The goal of the solution is to improve efficiency by transferring data storage from the blockchain to an external database. Less data is uploaded to the block chain, making computations on it more efficient.

The HLF blockchain's public record can only hold so much information. As the distributed ledger of the blockchain network gets bigger, the HLF's performance degrades. The recommended method uses a distributed ledger to keep track of the data's origin. Information are transferred into the Hadoop ecosystem.v by comparing them to data stored in the Hadoop database and data recorded in the distributed ledger. To expedite these procedures, each HLF peer node has a chain code in place. Additionally, the information such as checksum and provenance details are sent using a built-in client library. Thus, there is no longer a need for storage facilities for files managers. The blockchain is utilized to confirm that the info is accurate, while Hadoop allows for faster data processing than storing data there. The position and location of the data can be obtained from the ledger, and the specifics can be obtained by querying the Hadoop data repository.

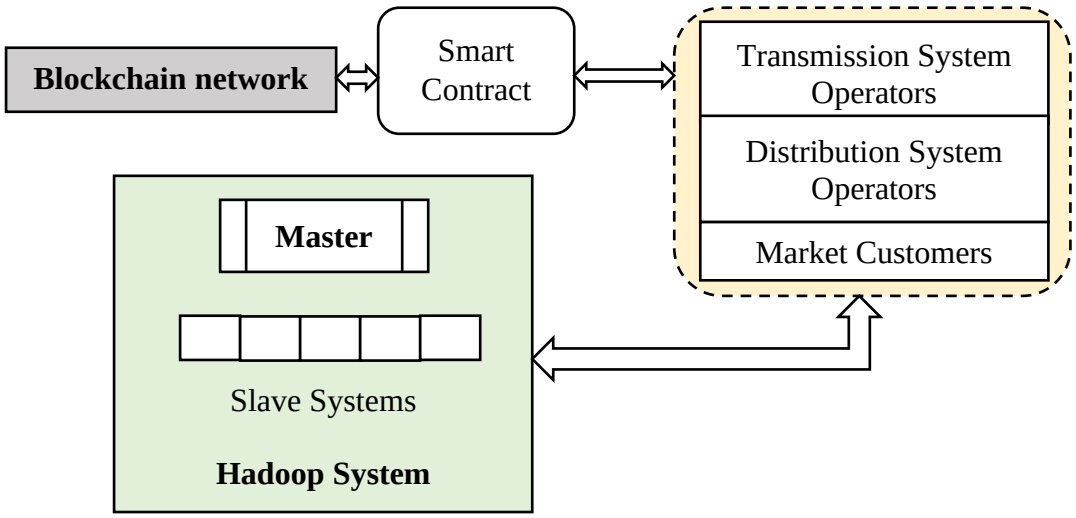

Figure 2: Data exchange system overblock chain's architectural design

Requests from user's details from a data source is assessed to determine whether that individual has the necessary rights. If the proposal is approved, the intelligent contract will carry out some preliminary checks. The user is then given the data after completing these tests. The suggested framework enables data exchange between smart grid users while keeping data security.

3.1 Conceptual system architecture overview

The suggested blockchain network's performance enhancement utilising ANN is described in this section. Figure 4 shows the ANN-based performance improvement's idea design. In order to guarantee network stability, The network known as blockchain is composed of up of multiple nodes that keep replicas of the public ledger and function as servers for smart contracts. The external, blockchain-connected ANN-based prediction module is available. Users of the network can submit transactions by using the smart contract's functions. Real-time benchmark results are tracked by the network, and the smart contract receives these numbers. The blockchain network's effectiveness is raised by using the ANN module. The network as a whole comes to a consensus, and the user receives the execution results. In the projected result DB, the predicted throughput values are persistent. Following each test, the network administrator is given these values in order to modify the network setup in accordance with the anticipated throughput and latency. The test is terminated when the ideal circumstances are present.

3.2 Computational model of ANN (Artificial Neural Network)

An Artificial Neural Network is a computational model that mimics the structure and functionality of the human brain. It consists of a vast network of interconnected functional units, or neurons, which work together to solve a particular problem [39]. Weighted directed graphs are used in the construction of ANNs. These graphs show the connections between the inputs and outputs of artificial neurons as nodes connected by directional weights and arrows. Based on their design, artificial neural networks (ANNs) can be divided into two groups: recurrent networks and fed-forward networks. A fed-forward ANN is chosen in this paper due of its enormous potential. Numerous configurations are evaluated to find the optimal ANN training module, including learning amount, activation function and learning rate. Several testing cycles were conducted for each config of the network training, and the mean results were stored for a later examination of the chance factor is employed to establish the ANN network's weights.The learn-to-predict framework overall Artificial Neural Network (ANN) model is depicted in Figure 5, as described in [39**].**

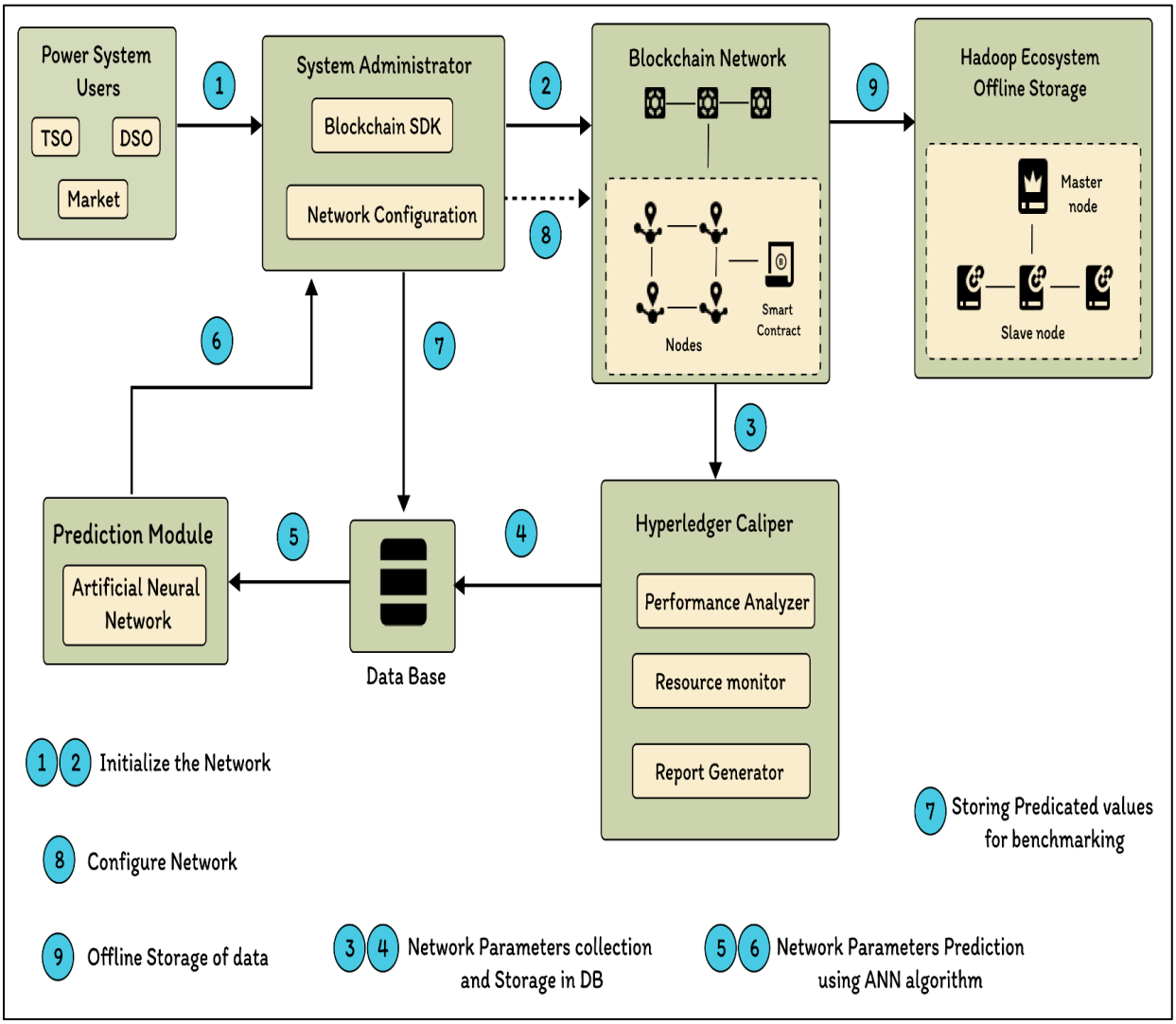

Figure 3: Block chain speed enhancement using ANN technology

The initial steps in the effectiveness optimisation process using an ANN include looking for missing values, importing the data, going through the data's fundamental preprocessing, and providing it with a data description. Thereafter, labels for optimal and non-optimal classes are assigned. The data are divided into three groups before being fed into the network: 15% are used for testing, 15% are used for validation, and 70% are used for training. Twenty neurons make form the hidden layer of the training network, which has 20 inputs and two outputs. Finding the best solution for the network requires continuously running the optimization module and analyzing benchmark results. The system administrator is then given these statistics so they can alter depending on the anticipated speed and latency, the system specifications. Execution of the learn-topredict model takes place apart from blockchain network. Equation offers a mathematical representation of how an ANN (Artificial Neural Network) learns to predict (1). The formula can be viewed as a function that moves through each layer of the network using an amount of weighed computations and activations to create an output. Mathematically, the outcome of an ANN given an input vector of x is expressed as:

$$y = f(Wn * f(Wn - 1 * \ldots f(W1 * x + b1) \ldots + bn - 1) + bn) \quad (1)$$

In the context of neural networks, the activation function can be represented by the letter "f." Additionally, weight matrices, denoted as Wn, Wn-1,..., W1, and bias vectors, denoted as b1,...," bn," are used in these networks. The value of n represents how many levels there there in the network.

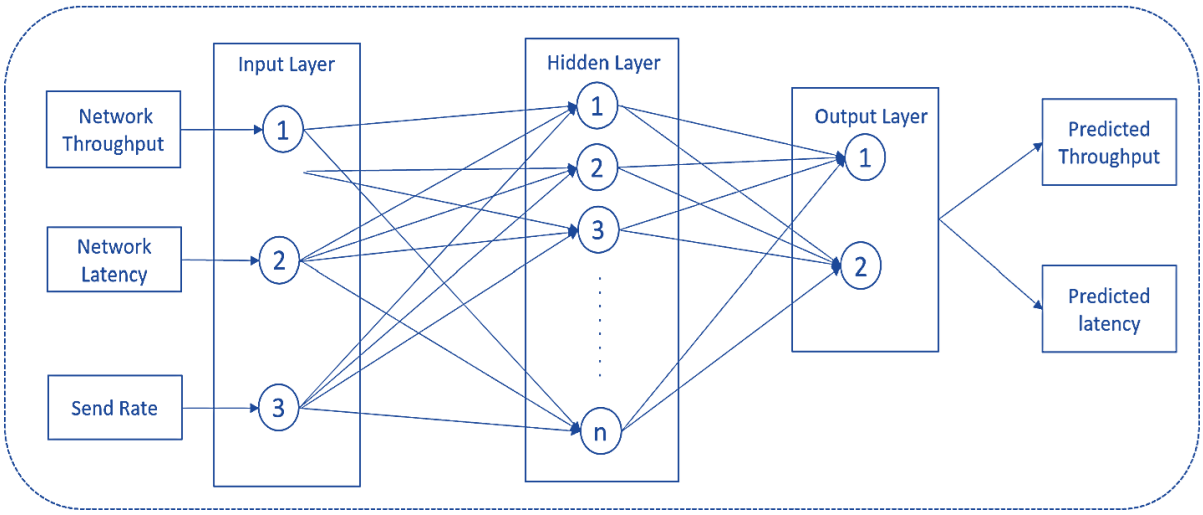

Figure 4: The detailed architecture of the point-to-predict ANN model

3.3 Use Case Implementation

This research paper highlights a specific communication method used by (TSOs) and (DSOs) during their operations. The Business Use Case (BUC) focuses on enabling the coordination of long-term network expansion plans related to smart grids at the interface between TSOs and DSOs. For the continued availability and reliability of the grid, it is vital to have effective long-term strategies for network expansion and reinforcement, which require successful information exchange and collaboration between TSOs and DSOs. When DSOs and TSOs discuss and decide on improvements, the network model simplifications that are already in place may be improved in the future. High-voltage power lines and interface substations may be added or removed as part of these modifications, and the TSO or DSO may also have plans to fortify the network. These plans might also take into account how important grid users are connected to DSO or TSO networks. This data is normally exchanged inbetween the two parties whenever the DSO/TSO interface plan is altered. The ideal time for deployment can be chosen by taking into account both the DSO and TSO network plans in order to find areas of optimization and synergy.

Figure 6 illustrates the application of the platform in the context of expansion plans. The TSO and DSO exchange vital information, including transmission and distribution network construction plans, using a four-step process facilitated by the platform. A detailed breakdown of the different stages involved in the use case is presented in Table 1.

Figure 5: Use case methodology

Table 1. Table of main differences involving blockchain technology and standard platforms.

Features	BlockchainPlatform	ConventionalPlatform
DataLoading	Distributed	Central platform
Accessing Data	Public	Central platform
ConsensusAppliance	DispersedConsensus	CentralDecisioning
Transparency	Noticeable	Partialvisibility
Trustworthy	Partial	HighlyScalable
Scalability	Consensus based	Central paltform

3.4 Stacked model

To sequentially describe the proposed volatility forecasting model, this section has been broken into many subsections. The first two sub-sections of the Stacked-Aritifical Neural Network model, which consists of two stages, are dedicated to that level's input data and algorithms, The third and fourth subsections of the text focus on the data required for the construction of the stack process, and they provide detailed information about the Artificial Neural Network (ANN) that was trained using the information discussed earlier. (Figure 1 provides a concise explanation of the method utilised for Stacked-ANN model estimate and testing.)

3.4.1 First class: Input data

Building the database with the explanatory variables selected to match the algorithms and Choosing a variability proxy to act as a response is the first stage.The True Realized Volatility (hereafter TRV) will be employed as the reaction element because the study's goal is to forecast future volatilities (Roh, 2006):

$$TRV_t = \sqrt{\frac{1}{n}\sum_{i=1}^{n} (r_{t+i-1} - \hat{r}_t)^2} \quad (2)$$

where$n = 5$ and$\hat{r}_t = \sum_{i=n}^{n} (r_{t+i-1})/n$.The size of the window used to compute the True Range Volatility (TRV) was selected carefully to ensure that it was large enough to generate precise results while also being small enough to minimize the mixing of different volatility regimes as much as possible.The input variables provided to the initial stage algorithms for predicting the True Range Volatility (TRV) consist of the thirty most recent volatilities calculated using returns that have been noticed in the market.

$$V_t = \sqrt{\frac{1}{n}\sum_{i=0}^{n-1} (r_{t-n+i} - \hat{r}_t)^2} \quad (3)$$

Where$n$= 5 and $\hat{r}_t = \sum_{i=0}^{n-1} (r_{t-n+i})/n$. Given the residual nature of the connections between earlier volatilities, the TRV, and the explanatory power of these correlations, it was decided to only consider the most recent thirty volatilities. As advised by Hastie et al. (2009), the historical data in order to assist in the training of the algorithms, will be adjusted to the range [0, 1] to 7. The component of the R project's quantmod is used to access the historical data (Ryan and Ulrich 2017). (R Core Team 2017).

It is significant to note that the figures for the first quarter are employed to fine-tune the first-tier algorithms before they are discussed . Following that, the second 50% of the data is employed for the Artificial Neural Network (ANN) computation, while the third quarter is reserved for testing purposes. A fresh data set that includes details for the upcoming year will be used to compare the accuracy and risk measurement of the benchmark models with the suggested one (e.g. If the Stacked-ANN model is tested and trained using data from 2000 to 2007, Market moves in 2008 would be the out-of-sample statistics used for comparison).

3.4.2 First Class: Individual algorithms

Below is a summary of the methods used to fine-tune the hyperparameters of the first-level algorithms in the Stacked-ANN structure:

- To train the first-level algorithms, it is necessary to minimize the Mean Square Error (MMSE) of the entire dataset.

- Block Bootstrap in a circle (CBB). By picking random blocks from the initial database, this technique (Politis and Romano 1991) creates fresh examples. These blocks' lengths are fixed, and Politis, White, and Patton (2004) developed the calculation method. (2009). Only fixed time series can be used with CBB.
- The Stationary Bootstrap (SB) is a time series analysis method introduced by Politis and Romano in 1994. This technique is similar to the CBB in that it can only be used for stable time series, but it differs in that the length of each block is chosen at random using a specific percentage that may be determined using various methods, as opposed to being pre-determined. (Patton et al., 2009; see also Politis and White, 2004).
- The (MEB) is a time series analysis method developed by Vinod in 2006 and later expanded upon by Vinod and de Lacalle in 2009. In contrast to the two previously mentioned techniques, the MEB generates new instances by drawing from dispersion of the underlying time series' maximum entropy, eliminating the requirement for stationarity.
- By removing the h data points between the explanatory variables and the reaction Marron's and Chu (1991) H Cross-Validation (HCV) method seeks to reduce the impact of any relationship that may exist between the factors explaining and the response it when working with time - series data. Lowering the actual self-correlation between the variable and response under examination yields the bandwidth, which has a maximum width of 100 days.

The process of grid search is employed to identify the most favorable combinations of hyperparameters for each of the five methods discussed earlier. These combinations are then tried against data not from the sample to determine which is the best exact choice for fitting the algorithm (the remaining 50% of the database).

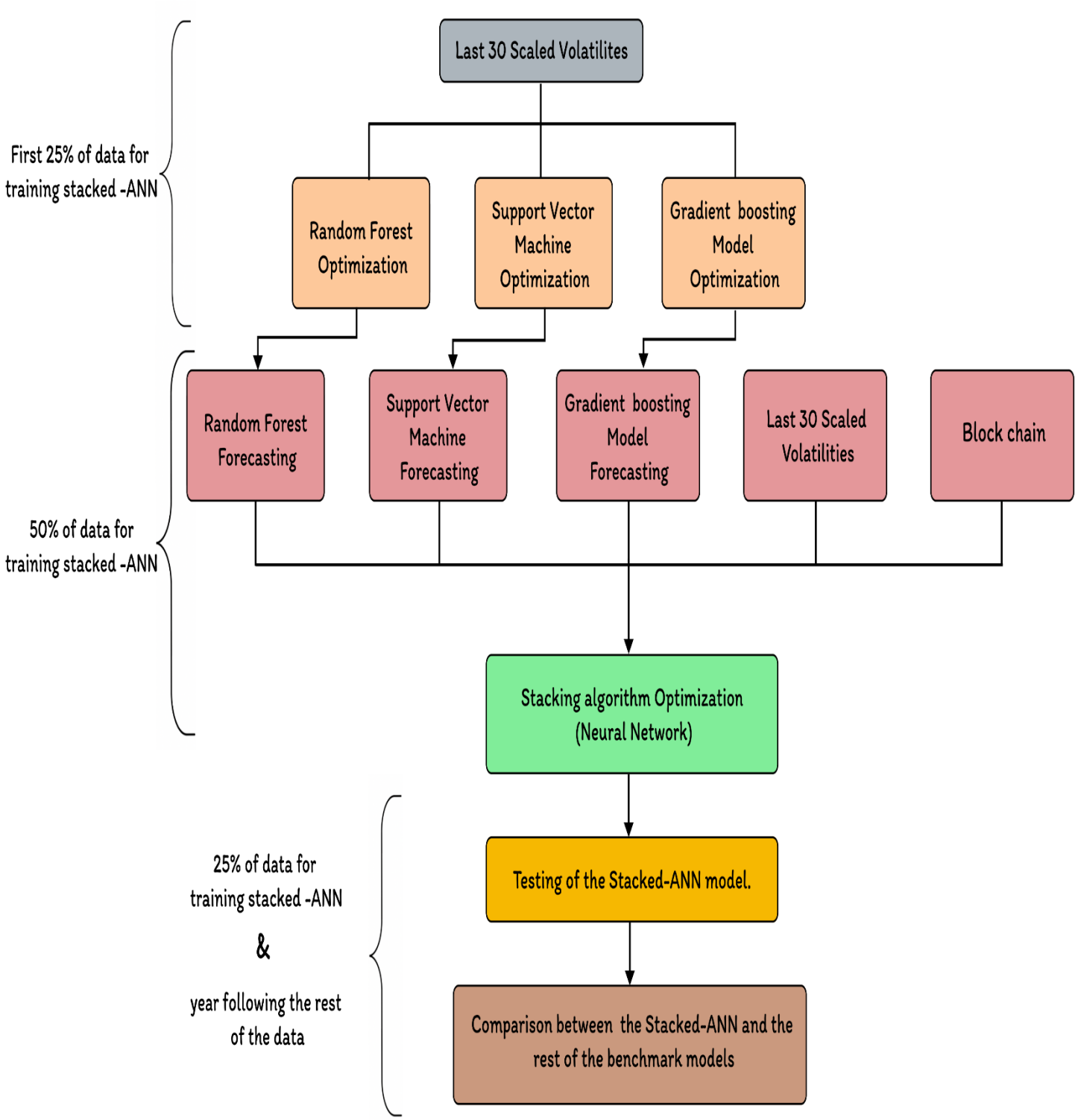

Figure 6: Structure of the stacked-ANN model working with blockchain

Grid search is applied to determine the optimal combination of hyper-parameters for each of the five methods described earlier. Once the best set of parameters is obtained through grid search, the model's performance is evaluated on the remaining 50% of the dataset. Finally, the option that best matches the algorithm's performance is chosen.

3.4.3 Second level: Input data

The ANN model is trained in three stages as described in Section 3.1. During the first stage, First-level algorithms are trained using the first 25% of the data. In the second stage, the ANN is trained on the following 50% of the data, and in the final stage, the remaining 25% of the data is utilized to evaluate the performance of the ANN. The explanatory variables provided to the ANN are listed below:

- The latest 30 volatility (V t,V (t-1),...,V (t-29)) algorithms were adjusted for the spread [0, 1] in a manner similar to the initial level algorithms.
- The first-level algorithm employs various techniques to forecast the True Realized Volatility, including Random Forest$\left(\widehat{TRV}_{t,RF}\right)$, Graient boosting $\left(\widehat{TRV}_{t,GB}\right)$ and Support Vector Machine $\left(\widehat{TRV}_{t,SVM}\right)$.

The response variable, as described in section 3.1, is the $\widehat{TRV}_{t,}$.

3.4.4 Second class: Stacking algorithm

The ultimate step of the Stacked-ANN model involves training the ANN by merging the predictions made by RF, GB, and SVM, as previously mentioned. It is important to note that the same techniques and procedures used for optimizing hyperparameters in the first-level algorithms are used here as well. This involves grid search combined with the methods described in Section 3.2, and selecting the final hyperparameters using the remaining 25% of the data. The following lists the basic traits and specifics of the stacking method:

- Twenty and ten neurons each in two hidden levels each make up the feed-forward ANN. All of the neurons in Using the function for sigmoid activation, the layers that are hidden have been turned on, whereas the single neuron-strong output layer has been activated using the linear activation function.
- The algorithm for optimization chosen is called ADAM (Adaptive Moment Estimation), and Kingma and Ba were the ones who developed it in 2014. This approach involves gradually adjusting while accounting for the early learning rate account both the most recent and earlier gradients. The authors' suggested default calibration is applicable when: $\beta_1 = 0.9$ and $\beta_2 = 0.999$.
- The group size corresponds to the amount of data needed to train the ANN, and there are 10,000 epochs in total.
- Calculations for the backward pass are made in accordance with the choice of the loss function using the root mean squared error.
- As explained in Section 3.1, the ANN is trained using 50% of the data, while the remaining 25% is reserved for testing. It should be noted that the first-level algorithms are calibrated using the initial 25% of the data.
- For the prediction model, the hyperparameters that need optimization are the level of L2 regularization parameter () and initially learning at a rate () used in ADAM.

The Stacked-ANN model (S-ANN) predicts the TRV t by utilizing the input data and the following equation:

$$\widehat{TRV}_{t,S-ANN} = \hat{f}\left(\widehat{TRV}_{t,RF}, \widehat{TRV}_{t,GB}, \widehat{TRV}_{t,SVM}, V_t, V_{t-1}, \dots, V_{t-29}\right) = \\ = h^{(3)}\left(\sum_{k=1}^{10} w_{1,k}^{(3)} h^{(2)}\left(\sum_{j=1}^{20} w_{k,j}^{(2)} h^{(1)}\left(\sum_{i=1}^{33} w_{j,i}^{(1)} x_i + w_{j,0}^{(1)}\right) + w_{k,0}^{(2)}\right) + w_{1,0}^{(3)}\right) \quad (4)$$

As stated in Section 3.3, xi represents the first level algorithms' predictions as well as the last 30 volatilities adjusted to the region [0, 1].

4. Experimental results

This section contains the experimental findings and discussions. Table 3 shows a comparison with current studies using conventional metrics for classifier evaluation.The traditional approach has multiple shortcomings. Firstly, it lacks detailed information about recognising positive and bad results, which is essential for locating ransomware. Secondly, it does not provide insights into the success and failure rates of the classification. Finally, it does not consider the net benefits of a specific classification model. To overcome these limitations and enhance

the identification and prediction of new and existing ransomware families, new evaluation metrics have been proposed in addition to the conventional ones.

LR (Likelihood ratio): In order to calculate the probability of an API pattern being associated with ransomware, the likelihood ratio (LR) [45] is utilized. Two types of LR are available: positive LR (PLR) and negative LR (NLR). A PLR higher than 10 and an NLR lower than 0.1 are recommended for optimal differentiation between ransomware and benign software. This helps to ensure good contrast between the two.

Diagnostic odds ratio (DOR): DOR is calculated by dividing the PLR by the NLR, resulting in a value that can be any number between zero and infinity. Typically, if the DOR exceeds 100, it is considered capable of distinguishing between ransomware and benign software. [46].Youden's index (Y):To evaluate the predictive accuracy of a model, Youden's index (Y) is used, which estimates a faultless classifier model for Y=1 that has neither false-positive nor false-negative results.

NND(Number needed to diagnose): The NND metric helps determine the bare minimum of data points required to make an accurate positive prediction [46]. A smaller NND value indicates better predictive performance according to the classifier model.

Table 2: Measures of performance for classifier assessment

ConventionalMetric	**Definition**
TPR(truepositiverate)	The true positive rate (TPR) is calculated as the number of true positive results divided by the sum of true positive and false negative results.
FPR	False positive rate (FPR) can be defined as the ratio of false positives to the total number of negative instances, where negative instances are those that do not belong to the target class.
Precision	The ratio of true positive to the sum of true positive and false positive.
Recall	The ratio of the number of true positive results to the sum of true positive and false negative results.
Accuracy	(TPS+TNs)/(TPS+FPS+FNs+TNs)
F-scoreordicecoefficient	2*Precision*Recall/(Precision+Recall)
PLR	TPSR/(1-TNs)
NLR	(TPSR-1)/TNs
DOR	PL/NL
NND	$1/[TPSR-(1-TNR)]^{1/Y}$,forY=1,NND=1/Y
NNM	NNM=1/Inaccuracy,where Inaccuracy=(FP+FN)/(TP+FP+FN+TN)
Y	TPR-TNR-1
NB	The expression (TP/n)-[(FP/n)*(p/1-p)] involves the variables n, which represents the total number of data points, and p, which is the probability threshold ranging from 10% to 99%. The first term in the expression is the ratio of true positives (TP) to the total number of data points (n). The second term in the expression is the product of the ratio of false positives (FP) to the total number of data points (n) and the ratio of the probability threshold (p) to the complement of the probability threshold (1-p).
EfficiencyIndex(EI)	The expression (TP+TN)/ (FP+FN) calculates the Efficiency Index (EI) of a binary classification model. The numerator is the sum of true positive (TP) and true negative (TN) classifications, while the denominator is the sum of false positive (FP) and false negative (FN) classifications. The Efficiency Index indicates how well the model is performing in terms of correctly classifying instances.

	The value of the Efficiency Index ranges from 0 to infinity, with a value of 0 indicating a completely inaccurate model and a value of infinity indicating a perfect model. In other words, the higher the Efficiency Index, the better the model is performing.

Table 3: Comparison with traditional performance metrics

Algorithm	Class	TPR	FPR	Precision	Recall	Accuracyin %	F-Score	Training time ins econd	Testing time in second
DecisionTable [35]	multiple	0.94	0.013	0.925	0.94	92.98	0.926	103.0	-
PART [35]	multiple	0.97	0.008	0.958	0.97	96.02	0.957	1609.0	-
SNN+ODT[36]	multiple	-	-	0.995	0.994	99.50	99.36	-	-
LSTM[37]	2.0	-	-	-	-	98.0	-	-	-
J48 [38]	2.0	-	-	-	-	97.2	-	-	-
CNN[39]	2.0	-	-	-	-	97.2	-	-	-
RF[40]	multiple	-	-	-	-	84.0	-	-	-
RF[41]	multiple	-	-	-	-	95.8	-	-	-
LSTM[42]	multiple	0.97	0.027	-	-	-	-	-	-
SVM[43]	multiple	-	-	-	-	-	-	-	-
GTB[44]	multiple	-	-	-	-	-	-	-	-
ResilientKNN, k=99optimal	multiple	0.987	0.899	1.0	1.0	-	1.0	150.47	10.35
Proposed model+FLR(proposed)	multi	1	0	1	1	88.7	1	135.24	3.98
Proposed +Resilient KNN, k=10optimal(proposed)	multi	1	0	1	1	97.3	1	346.69	5.52

Table 4: Comparison using a novel metric with related works

Algorithm/Metric	PLR	NLR	DOR	Y	NND	NNM	NB10%	NB50%	NB99%	EI
TDA[23]	0.23	-	-	-	-	-	-	-	-	-
DBSCAN[23]	0.06	-	-	-	-	-	-	-	-	-
XGBoost[23]	0.0	-	-	-	-	-	-	-	-	-
RandomForest[23]	0	-	-	-	-	-	-	-	-	-
ResilientKNN(ours)	1.09	0.14	7.785	0.086	11.627	52.63	0.108	0.973	96.327	49.44
FuzzyLogisticReasoning(FLR) -Proposed	∞	0	∞	1	1	37.03	0.032	0.291	28.866	33,052
ResilientKNN -Proposed	∞	0	∞	1	1	8.849	0.032	0.291	28.866	7,627

NNM (Number needed to misdiagnose): A model's ability to make predictions should be assessed, the minimum number of data points needed to make an incorrect prediction, known as NND, must be determined. Conversely, the minimum a model's ability to make predictions should be assessed to make accurate predictions, known as NNM, is recommended for improved performance [46].Net benefit (NB): Based on a threshold probability or cutoff point for different exchange prices varying from 10% to 99%, the Net benefit gives information about the correct or incorrect classification [47].

Table 4 presents a comparison with prior research using novel metrics.

Based on the experimental results, it was found that all types of ransomware have PLR values exceeding 100 and NLR values equal to zero, It shows that the classification model has significant positive and negative probabilities. Additionally, the DOR has an unlimited value, suggesting that the suggested models can successfully differentiate between goodware and ransomware. The models' effectiveness is further demonstrated by the fact that Y=1 for both models, indicating that they have never made a misclassification. Moreover, the NND value of 1 and NNM values of 37.03 and 8.849 for Resilient FLR and KNN, respectively, demonstrate that both models can effectively identify goodware and ransomware groups. In comparison to other researchers' work, our proposed models, Resilient FLR and KNN with proposed model, outperform DBSCAN [23] and TDA [23], they are more effective at producing reliable predictions since they have an unlimited PLR value.

The outcomes presented in Table 4 indicate that the proposed model combined with Resilient KNN FLR models outperformed all other models in terms of the highest PLR, zero NLR, infinite DOR, NND=1, Y=1, and net benefit (NB) for three different ranges of 10%, 50%, and 90%. Specifically, the PLR values for these models were 0.032, 0.291, and 28.866, respectively. It is noteworthy that the proposed model combined with Robust KNN outperformed its FLR equivalent in terms of NNM, with a value of 37.03 compared to 8.849 for FLR.Table 4 shows that the FLR with proposed model has the highest Efficiency Index (EI) number of 33052 when compared to other models, making it the most effective model.

5. Conclusion and futuristic work

An innovative network for smart grid data exchange based on blockchain operators is introduced in the article. A learn-to-predict ANN model has been added to improve performance. The proposed method has been evaluated by developing a use case for data exchange within smart grids on the Hyperledger Fabric, a distributed network. The majority of current platforms for exchanging information and data are centralised, which frequently results in security flaws like the possibility of single points of failure, malicious attacks, and manipulated data. Blockchain's decentralised consensus method can assist in resolving these problems. The suggested platform has the potential to increase TSO-DSO interoperability and the system's overall efficacy in terms of congestion and supply security control by enabling trustworthy data exchanges. Beyond smart grids, the platform's possible uses include various fields that call for user-to-user information and data interchange, such as smart cities designed to support the long-term viability of bitcoin network.

Results show that applying the output of the proposed models resulted in categorization accuracy of 88.7% and 97.3% respectively. The study also proposed seven new assessment measures to evaluate the models' effectiveness, containing the probability ratio, Youden's index, diagnostic odds ratio, number needed to correctly diagnose and correctly misdiagnose, net advantage, and effectiveness index.Comparing the proposed models to previous research using these metrics, it was found that both resilient FLR and KNN models performed similarly well when compared to DBSCAN, TDA, Random Forest, and XGBoost trees that have significant PLR. However, the resilient KNN model outperformed FLR due to its higher NNM value, indicating fewer incorrect predictions. On the other hand, FLR was found to be more effective than robust KNN in terms of efficiency index.In order to provide assistance to researchers, practitioners, and the general public in making decisions, the study aims to create new methods and datasets for the future identification and forecasting of crypto-ransomware.

References

[1] S. Ruoti, B. Kaiser, A. Yerukhimovich, J. Clark, and R. Cunningham, “Blockchain technology: what is it good for?” Communications of the ACM, vol. 63, no. 1, pp. 46–53, 2019.

[2] Bai, D. P., & Preethi, P. (2016). Security Enhancement of Health Information Exchange Based on Cloud Computing System. International Journal of Scientific Engineering and Research, 4(10), 79-82.

[3] S. Kok, A. Abdullah, N. Jhanjhi, and M. Supramaniam, “Prevention of crypto-ransomware using a pre-encryption detection algorithm,” Computers, vol. 8, no. 4, p. 79, 2019.
[4] Preethi, P., & Asokan, R. (2020, December). Neural network oriented roni prediction for embedding process with hex code encryption in dicom images. In Proceedings of the 2nd International Conference on Advances in Computing, Communication Control and Networking (ICACCCN), Greater Noida, India (pp. 18-19).
[5] Asokan, R., & Preethi, P. (2021). Deep learning with conceptual view in meta data for content categorization. In Deep Learning Applications and Intelligent Decision Making in Engineering (pp. 176-191). IGI Global.
[6] Akcora, C. G., Li, Y., Gel, Y. R., & Kantarcioglu, M. (2020).BitcoinHeist: Topological Data Analysis for Ransomwareprediction on the Bitcoin blockchain. Proceedings of theTwenty-Ninth International Joint Conference on ArtificialIntelligence. https://doi.org/10.24963/ijcai.2020/612
[7] Liao, K., Zhao, Z., Doupe, A., & Ahn, G. (2016). Behind closed doors:measurement and analysis of CryptoLocker ransoms inBitcoin. 2016 APWG Symposium On Electronic CrimeResearch (Ecrime). doi: 10.1109/ecrime.2016.7487938.
[8] D. Y. Huang et al.(2018). Tracking Ransomware End-to-end.IEEESymposium on Security and Privacy (SP).pp 618-631, doi:10.1109/SP.2018.00047.
[9] Alhawi, O. M., Baldwin, J., & Dehghantanha, A. (2018). Leveragingmachine learning techniques for windows Ransomwarenetwork traffic detection in Cyber threat intelligence.Springer, Cham. pp. 93-106. doi: 10.1007/978-3-319-73951-9_5
[10] Kshirsagar, D., & Shaikh, J. M. (2019, September). Intrusion DetectionUsing Rule-Based Machine Learning Algorithms. In 2019 5thInternational Conference On Computing, Communication,Control And Automation (ICCUBEA) (pp. 1-4). IEEE.doi: 10.1109/ICCUBEA47591.2019.9128950
[11] Hussein, N., Abbas, A., & Mahdi, B. (2021). Fraud Classificationand Detection Model Using Different Machine LearningAlgorithm. Tech-Knowledge Journal, 1(1).
[12] Sohail, M. N., Jiadong, R., Muhammad, M. U., Chauhdary, S. T.,Arshad, J., & Verghese, A. J. (2019). An accurate clinicalimplication assessment for diabetes mellitus prevalencebased on a study from Nigeria. Processes, 7(5), 289.doi:10.3390/pr7050289
[13] Gaikwad, D., & Thool, R. (2015). Intrusion Detection System UsingBagging with Partial Decision TreeBase Classifier.Procedia Computer Science, 49, 92-98. doi:10.1016/j.procs.2015.04.231
[14] Alam, M., Ubaid, S., Shakil, Sohail, S., Nadeem, M., Hussain, S., &Siddiqui, J. (2021). Comparative Analysis of MachineLearning based Filtering Techniques using MovieLensdataset. Procedia Computer Science, 194, 210-217. doi:10.1016/j.procs.2021.10.075.
[15] Preethi, P., Asokan, R., Thillaiarasu, N., & Saravanan, T. (2021). An effective digit recognition model using enhanced convolutional neural network based chaotic grey wolf optimization. Journal of Intelligent & Fuzzy Systems, 41(2), 3727-3737.