International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING

ISSN:2147-6799 www.ijisae.org Original Research Paper

Proof of Learning based Block Chain with Progressive Conditional Generative Adversarial Network espoused Fake Check Scams Detection and Prevention

Mr. SrinivasaVarma M[1*], Dr. Pardha Saradadhi Varma G[2], Mrs. Hemalatha I[3]

Submitted: 28/01/2024 **Revised**: 06/03/2024 **Accepted**: 14/03/2024

Abstract: The issues of fraud and other irregularities in Bitcoin network are discussed in this paper. These are typical issues with online transactions and e-banking. But fraud and anomaly detection techniques also change as the financial industry does. Additionally, blockchain technology is being presented as the safest approach to be included into finance. However, a lot of frauds are also rising annually along with these sophisticated technologies. Therefore, proposed a method proof of learning based Block Chain with Progressive conditional Generative Adversarial Network for Detecting and Preventing Fake Check Scams (BC-PCGAN-EFDB). Proof of learning based Block Chain is utilized. More specifically, a Block Chain technique based on proof of learning makes it possible to confirm a check's legitimacy without disclosing personal information about the bank's clients. The Cashing-Bank may choose to proceed with the transaction or to terminate it after this verification. Furthermore, the Block Chain technique based on proof of learning has no effect on the current bank's protocols for verifying the legitimacy of checks. In order to enhance the identification and avoidance of Fake Check Scams and reduce blockchain latency, a Progressive conditional Generative Adversarial Network (PCGAN) is suggested. Here, the BC-PCGAN-EFDB proposed approach is implemented and the performance metrics, like classification error, precision, accuracy, true positive rate, computational power, integrity, availability and Confidentiality are analyzed. The proposed method gives higher accuracy 20.76%, 15.98% and 14.78% and higher precision 23.78%, 30.98% and 15.67% when comparing with existing techniques like machine learning and block chain based efficient fraud detection mechanism (ML-BBEFDM), credit card fraud detection utilizing block chain and simulated annealing k-means algorithm (CCFD-BC-SAKA) and blockchain-based solution for detecting and preventing fake check scams (BC-DFCS), methods respectively.

Keywords: *Block Chain, Progressive conditional Generative Adversarial Network, Proof of learning, Probability-Based Synthetic Minority Oversampling Technique.*

1. Introduction

The development of technology has led to modernization across all sectors, including banking, education, healthcare, and others. Additionally, as communication technology has advanced, so too have internet purchases and payment methods. Traditional currencies are being transformed into digital currencies as a result of this modernization, and all financial transactions are now carried out online. These transactions, however, lack complete security and are susceptible to numerous cyber-attacks, including privacy violations, abnormalities, and fraud problems. Additionally, there is an increase in financial transaction fraud as the amount of transactions rises. As a result, there are annual losses of billions of dollars worldwide [1]. An anomaly is any unusual behavior on a network that performs strangely. Anomaly detection is used in cyber security and digital financial exchange to find fraud and network intrusion. Anomaly detection aims to safeguard network from fraudulent, illegal actions. Applications for anomaly detection examined at strange activities in the financial sector and found hackers and dishonest users. But in conventional financial systems, every approach of anomaly detection is created for centralized systems [2-4]. So, as digital currencies like Bitcoin grow in popularity, anomaly detection techniques based on the blockchain are getting better. Even with these improvements, fraud still happens often. There is no viable solution for centralized schemes, despite fact that numerous AI, ML approaches offered to detect abnormalities, fraud in digital transactions. In many industries, blockchain technology is the most advanced and is developing swiftly. It initially emerged into the public eye with the launch of Satoshi Nakamoto's Bitcoin in 2008 [5, 6].

It tackles the security problems with centralized systems and offers defenses against outside dangers. All records are time stamped, and the ledger is distributed, decentralized, and immutable to guarantee record integrity. However, a small number of blockchain network users engage in malicious behavior [7]. The financial industry is the one most impacted by cybercrime, which is growing daily along with technological advancements [8]. Financial

[1]*CTO, Director, Sri Maharshi Consultancy Private Limited, Scholar, KL University, India.*
[2]*Department of Computer Science and Engineering Vice Chancellor, Koneru Lakshmaiah Education Foundation, India.*
[3]*Professor, Department of Information Technology, S.R.K.R. Engineering College, Bhimavaram, India.*
** Corresponding Author Email: varma@srimaharshiconsultancy.com*

systems' security flaws are the primary cause of this issue. These systems have anomalies, which are also referred to as frauds. The most frequent frauds in traditional financial schemes are credit card frauds, which resolved with assistance of AI approaches. These frauds cause the financial sector to lose billions of dollars annually as a result [9]. The authors of [10] used unsupervised machine learning methods to find the financial irregularities. On other hand, supervised ML methods work better for detecting fraud, according to [11]. Due to its decentralized and unchangeable nature, it offers security and anonymity to the financial industry. It does not, however, address problems like privacy invasion, Sybil assaults, and double-spending attacks. These attacks aim to increase financial benefits and deter illicit behaviors. Proof of work is the foundation of the digital currency known as Bitcoin (PoW). Utilizing digital signatures, hashes obtained through timestamp service, all digital transactions within the Bitcoin network are carried out in a distributed fashion. A reliable third party is not required for the verification of Bitcoin transactions. Consequently, user spend same currency twice, becomes fraudulent transaction, known as double-spending attack [12]. Fraud is big issue in business,bank fraud poses a serious threat to a bank's ability to grow organizationally. Specifically, one of the most significant issues facing financial organizations is counterfeit checks. The primary cause is that criminals can now more easily fabricate authentic counterfeit and fake checks thanks to technological advancements. Thanks to technological improvements, criminals can now commit creative scams that are very hard to uncover. It extremely difficult to discern erasable ink alterations, printed signatures on digital pictures because most banking schemes accept scanned copies of checks for approval. Considering that there isn’t already a check authentication technique that only uses IT resources [13, 14]. While all of the above techniques have the potential to identify counterfeit checks, they are out of date and only offer physical protection, rendering them useless in the current situation when bank clients are able to print their own checks. In order for a fake check detection system to be effective and, consequently, widely adopted, it must be developed in a way that satisfies the necessary requirements and is simple to integrate into the current bank equipment.

The main contribution of this paper is,

- In this research, proof of learning based Block Chain with Progressive conditional Generative Adversarial Network is proposed for detecting, preventing fake check scams.
- In this work, effective proof of learning based Block Chain [22] is proposed supports banks to share information about providing checks. Further exactly, proof of learning based Block Chain method supports to verify authenticity of given check, without revealing banks’ customers’ personal data. Following verification, Cashing-Bank can decide to continue transaction else abort it. Furthermore, proof of learning based Block Chain approach doesn’t affect existing bank’s procedures though checking authenticity of checks.
- Then further maximize Fake Check Scams detection and prevention and minimize the latency of blockchain, a Progressive conditional Generative Adversarial Network (PCGAN) [23] is proposed.
- Here, proposed approach executed, performance metrics like classification error, precision, accuracy, true positive rate, computational power, integrity, availability and Confidentiality.

Remaining part is arranged as below: section 2 describes literature review, section 3 explain proposed method, section 4 describes result and discussion and section 5 conclusion.

2. Related Work

Numerous study were suggested in the literature based upon Fake Check Scams Detection and Prevention, certain current works divulged in sector,

Ashfaq et al., [15] have presented secure fraud detection mechanism depend on ML and blockchain based efficient fraud detection mechanism. For transaction classification, dual ML methods are used: RF, XG boost. By using integrated and fraudulent transaction patterns to train the dataset, ML methods were able to anticipate future incoming transactions. To identify fraudulent transactions in Bitcoin network, machine learning algorithms are linked with blockchain technology. It uses the RF and XG boost processes to forecast transaction patterns and classify transactions. It has high accuracy but less recall.

Rani, et al., [16] have presented simulated annealing and blockchain technology, CCFD block chain technologies, simulated annealing are combined the research suggesting the k-means process to detect credit card fraud. A private-permission blockchain network, k-means was used in conjunction with simulated annealing to identify suspicious and aberrant banking transactions. According to experimental results, the suggested method outperforms the straightforward k-means algorithm in terms of accuracy. It also has high accuracy but less precision.

Hammi et al., [17] have presented a blockchain-based system to verify checks and identify fraudulent check schemes is being developed. Its goal was to prevent and detect fraudulent check scams. Additionally, permits revocation of previously utilized checks. Furtherexactly, allows banks to exchange details about both providing, utilized checks while protecting the privacy of their clients. It shows how the method, which makes use of Namecoin and Hyper ledger blockchain technologies, is proof of concept. It has high precision and also has high computational time.

Wang et al., [18] have described the concept of nudges applied to blockchain-depend data privacy management in open banking. A suggested strategy was novel data privacy management framework for banking industry based on blockchain technology. Three components make up the framework: novel collaborative filtering-depend method; data disclosure confirmation system for customer strategies depend on Nudge Theory; and a data privacy classification technique based on characteristics of financial data. A prototype was implemented, set of processes was proposed for framework. It has high precision but less sensitivity.

An et al., [19] have presented, some in the debate over the introduction of a central bank digital currency (CBDC) contend that digital currencies are only useful for boosting an economy's liquidity in times of market stress. Furthermore, decentralization and consensus are two ideas that are central to the majority of the benefits of blockchain applications that are now known. The decentralization of blockchain technology has the potential to democratize corporate governance, banking services, and the real estate sector. It also has high accuracy but less precision.

Morishim, [20] have presented method, which uses a graphics processing unit for scalable anomaly detection in block chain, sub graph-depend anomaly detection technique uses a portion of blockchain data for detection. Suggested sub graph structure makes use of parallel processing on GPUs to speed up detection. When there were one hundred targeted transactions in an evaluation utilizing actual Bitcoin transaction data, the suggested solution outperformed an existing GPU-based method by 11.1 xs without sacrificing detection accuracy. It also has high accuracy but less precision.

Kapadiya et al., [21], have presented healthcare insurance fraud detection utilizing block chain, AI: analysis architecture, Prospects. It provides taxonomy of different health insurance security challenges together with a methodical survey for blockchain-enabled safe health insurance fraud detection. To identify health insurance fraud, we suggested a safe, intelligent solution built on blockchain and artificial intelligence. Following that, a case study of health insurance fraud was given. The implementation of block chain with AI-powered health insurance fraud detection scheme presents its last unresolved concerns and research obstacles. It has high precision but less sensitivity.

3. Proposed Methodology

In this section describes the proof of learning based Block Chain with Progressive conditional Generative Adversarial Network for Detecting and Preventing Fake Check Scams (BC-PCGAN-EFDB). The two layers of the suggested system model are machine learning and blockchain. Transactions are started by the blockchain model and subsequently classified as malicious or legal using machine learning models. This categorization is binary. The suggested system model for fraud, anomaly detection in financial industry is predicated on the combination of blockchain technology and machine learning. The anomaly detection system finds suspicious, out-of-the-ordinary events that diverge from the majority of the data. For the suggested method, dataset of bitcoin transactions is utilized. To distinguish between malicious and genuine transactions, it additionally employs the Progressive conditional Generative Adversarial Network (PCGAN). Additionally, new incoming transactions are predicted by the classifier. Using the provided dataset, the suggested method trained, examined for both benign, malevolent data patterns. Following steps (explained in the subsections below) make up the suggested system model.

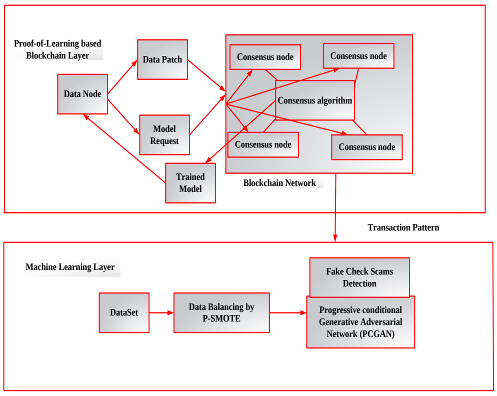

Fig 1: Proposed system mode of block chain with Progressive conditional Generative Adversarial Network (PCGAN)

3.1 Data balancing using by Probability-Based Synthetic Minority Oversampling Technique (P-SMOTE)

In ML, where distribution of classes highly unbalanced, data imbalance is a key issue. Unbalanced data reduces the accuracy of machine learning systems. It goes up in cases where one class has more instances than the other. P-SMOTE [22] is therefore utilized to address this issue, synthetic samples created at random for minority class. This method resolves over fitting issue brought on by the data's random oversampling. Data point chosen at random from minority class as the basis for this method. Next, its neighbors are given random weights, and these neighbors are included in the initial samples.

Algorithm1: Algorithm of P-SMOTE

```
Require: $S$ -minority class samples, $G$ -Synthetic examples, $F$ -feature space dimension,
$p$ - Probability distributions list, $pdf$ -probability density
```

```
function.
Ensure: $A$: Collection of new samples
$A \leftarrow \phi$;
for all $f \in F$ do
$Err = +\infty$
$hist = histogram(S_f)$ ;
$l_f \approx G(\nu, \sigma^2)$ ;
for all $(l \in L)$ do
Proper distribution $l \; to \; S_f$
$h = \sum^{S_f} (hist - pdf_l)^2$ ;
if $(h < Err)$ then
$Err = h$
$l_f = l$ ;
for all $(g \in G)$ do
        for all $(f \in F)$ do
        $x_f \approx l_f$
        $A \leftarrow A \cup x$
return $A$
```

3.2 Proof of learning based Block Chain

The effective proof of learning based Block Chain [23] is proposed that supports banks to share information about providing checks. Further exactly, proof of learning based Block Chain method supports to verify authenticity of given check, without revealing banks' customers' personal data. Following this verification, Cashing-Bank decide to continue transaction else to abort it. Furthermore, proof of learning based Block Chain approach doesn't affect existing bank's procedures though checking authenticity of checks.

Data nodes and consensus nodes are the two sorts of entities that make up decentralized peer-to-peer network that is subject of proposed system. Public-private asymmetric encryption techniques are used to secure communication between nodes, which takes form blocks that are broadcast to whole network. In suggested system paradigm, Figure 1 illustrates how the consensus and data nodes are generalized. A user who uses the blockchain to commission machine learning jobs is known as a data node. The consensus nodes provide the system with processing power, compete to train a method satisfies specifications given by data node. Data node's reward is given to winner node. Also providing method training prizes, block chain works as decentralized data repository of encrypted data, ordinate transfer transactions. It characterize major components of system as follow

The organizations known as consensus nodes, or computing power suppliers, get machine learning tasks from data nodes. A training dataset, desired ML method, minimal accuracy, reward are all included in a job that a data node issues. The training dataset is kept in blocks and encrypted. Hash pointers to training set, but not test set, are contained in a task request. The task is added global task list and broadcast to entire network at the time of request. One significant piece of content in most recent valid block (one before present block) is global task list. The prize is promptly moved from the account of data node to virtual reservoir account, used to compensate the consensus winner by best generalization performance. Only initial transactions of each block can receive block rewards (both block, ommer rewards) from this account; subsequent transfers from this account are deemed invalid.

It is possible to omit hyper parameters utilized for training, likes learning rate else weight decay coefficient, from method specification. The method specification comprises complete specification of network design, including number and types of layers, their interconnections. A comprehensive collection of model parameters should allow for the performance of inference based on the model definition, which should be sufficiently detailed. The standard also specifies a time restriction for training, an accuracy meter, minimum training accurateness that satisfied. To stop criminal consensus nodes exploiting test set training, it should stay off the blockchain. Consequently, once the data node begins receiving trained models, it is only then that it broadcasts the test set. Before releasing test set, data node may choose to hold off until several solutions have been received. There will be no acceptance of consensus node answers after the test set is made public. This is verified by comparing it to test set timestamp, signed by data node thwart forgeries.

The time difference between nodes in a distributed system, even when their clocks might not be exactly synchronized, far less than time required to train ML method . Each task's test data kept in current block, where consensus nodes take fought for the right to use their computing capacity to train the relevant task. Any node receiving novel valid block be able to validate suggested winning solution using test data pointers in block header. Through adding freshly received tasks, removing trained task, task list in valid block is transformed into global task list.

Consensus nodes, known as miners, providers of computing power to network and behave according to proof-of-learning consensus protocol algorithm 2. Miners compete to complete method training jobs provided by data nodes, rewarded as consequence.

Algorithm 2: Proof-of-Learning consensus protocol

```
Input: task-list: task list stored in previous block
Block-chain: the blockchain
PHS: hash value of previous block
CHS: hash value of current block
Output: block: new generated block
Step 1: Task –Popmost valuable (task-list)
Step 2: Train-data: collect data (task.data-pointers)
Step 3: SMLayer: Create SMLayer (PHS, CHS, task.model.input-length)
Step 4: Sm-model:Insert Layer (task.model, SMLayer)
Step 5: Received_blks-[]
Step 6: While t<time-max && not received test-data do
    Step 7: Train sm-model for one step
    Step 8: Calculate train accuracy
    Step 9: If train accuracy $\geq$task.required-accuracy then
        Step 10: Blk: createblock (sm-model)
        Step 11: Broadcast block to other consensus nodes
        Step 12: Append (received-blocks, block)
    Step 13: End if
    Step 14: If received novel solution block then
        Step 15: Append (received-block, block)
    Step 16: End if
    Step 17: $t++$
Step 18: End while
Step 19: If not received test-data then
    Step 20: Test-data=training-data
Step 21: End if
Step 22: Sort received –blocks descending order of test accuracy
Step 23: For all block in received-blocks do
    Step 24: If verifyblock (block,PHS, test-data) then
        Step 25: Append (block-chain, block)
        Step 26: Return block
    Step 27: End if
Step 28: End for
```

When a consensus node is idle, it chooses the job that has the highest value from list of tasks in most recent block of blockchain. This task serves as the best consensus task the node is aware of, and it starts training. Average reward in given amount of time determines task's value. Highest value task ought to same for every miner if prior block is consistent. Lines 2-4 describe the subsequent maintenance processes that it carries out, including data collection in accordance with the task description, model parameter initialization, and the creation of secure mapping layer, which converts cipher text input into feature vectors for task model training. Malicious nodes unable to begin mining before development of preceding block because generation of SML tied to both CHS, PHS.

Next, using a method of its choosing, miner optimizes given ML method (Step 7). The miner adds novel block to end of local chain and broadcasts a new block announcing its success when trained method reaches minimum training accurateness (Steps 9–13). If both the training time restriction, test data have not released, miner ensues train its task method (Line 6). Miner recognizes loses competition in block height, ends training if test data released before finishes training task. Miner might get additional fresh blocks from other miners who say they finished the assignment correctly before it finishes its own training. The miner saves these blocks and proceeds with training if it not yet received test data (steps 14–16). Maximum training time has elapsed but model requester has not, knowingly or unknowingly, delivered the test_data, it is possible. For verification purposes, training data considered test data in this instance (Steps 19–21). In this particular instance, the block winner will be miner who contributes initial block that meets necessary precision.

Any one of the following three scenarios will result in the miner ending its training session: One of three things has happened: (1) miner has identified method that meets minimal training accurateness requirements; (2) training period has run out; else (3) miner takes received test data, which indicates that no more method solutions will accepted. Consensus nodes sort blocks descending order of test accurateness after comparing their test accuracy (Step 22). This process occurs after they receive a sequence of blocks. The validity of each block is then assessed by miners using algorithm 2 (Line 29). This algorithm's computational time complexity O (1).Algorithm 2 returns true, block first passes verification procedure and becomes winning block (Lines 24–27). Remaining blocks that meet criteria are designated assumer blocks.

Algorithm 3: Proof-of-Learning verification process

```
Input: block: received new block
PHS: hash value of previous block
Test-data: the testing dataset
Output: verified : true or false
Step 1: CHS=hash (block);
Step 2: SMLayer1:Create SMLayer (PHS, CHS, 1)
Step 3: If SMLayer !=block.model.SMLayer (1) then
    Step 4: Return false
Step 5: Else
    Step 6; Test-accuracy: calculate accuracy (block.model, test-data)
    Step 7: If test-accuracy $\geq$ required-accuracy then
        Step 8: If block.timestamp<test-data.timestamp then
            Step 9: Return verified =true
        Step 10: End if
    Step 11: End if
    Step 12: Return verified =false
Step 13: End if
```

Test data is subsequently added to the block's body upon acceptance as a winning block, block owner receives task's rewards. Finished task is subtracted from initial list in previous block, afresh gathered tasks are along with create task list in following block. Transactions in winning block will be regarded as valid by consensus nodes, which will then attempt the subsequent task from the winning block's task list to create new blocks. Only transaction in winning block will be accepted by the whole network. By referencing these ommer blocks, the subsequent winning blocks can receive further prizes, and the ommer block's maker can also receive awards. The miner's blockchain is then updated with winning block, subsequent blocks.

How to verify a received block is demonstrated in Algorithm 3. First secure mapping layer created depend on PHS, CHS (steps 1, 2) prior to a miner verifying a block, verification rejected if computed secure mapping layer differs from that novel block. If not, miner uses publicly available test data (step 6) to confirm the test's correctness. The block passes verification if test accuracy higher than necessary accurateness (steps 7–11). If more than one block is verified simultaneously, training data finished before allotted time, accuracy satisfies task necessities, first verification-pass block declared winner, remaining blocks classified as ommer blocks. Block with highest accuracy is deemed the winner when training time surpasses maximum completion time; in this scenario, there isn't a middle block.

3.3 Progressive conditional Generative Adversarial Network (PCGAN)

Then further maximize the Fake Check Scams detection and prevention and minimize the latency of blockchain, a Progressive conditional Generative Adversarial Network (PCGAN) [24] is proposed. Fraud and irregularities in online systems are increasing as more firms move their operations online. Online fraud has been addressed through the use of fraud detection systems based static rules developed human specialists. Because of this, businesses have to reduce the amount of fraudulent activity that occurs during online transactions. We address fraudulent Bitcoin transactions in this study. Anomaly detection can be used to identify outliers, or unusual patterns that deviate from expected behavior. Dataset of bit coin transactions employed in suggested method. Financial sector's bit coin transactions serve as the foundation for this dataset. As far as we are aware, there are similarities between the transaction patterns of ethers and bit coin. As a result, we used the bitcoin dataset to train our algorithm, and it also accurately predicts ether transaction data. The mathematical expression of PCGAN is shown in bellow;

A PCGAN is a type of NN architecture utilized in the field of generative modeling, mainly in context of image generation. An extension of the traditional Conditional Generative Adversarial Network (CGAN) and also takes inspiration from the Progressive Growing of GANs (PGGANs).

Step 1: Data Preparation

Prepare a dataset of data’s with conditioning information.

Step 2: Generator Network

The generator network, denoted as G, takes random noise vectors (z) and conditioning information (c) as inputs to produce fake data’s $(x-fake)$. The generator aims to generate realistic data’s that match the given conditioning information. The generator network can be represented as in equation (1),

$$G(z,c)\rightarrow(x-fake) \tag{1}$$

Step 3: Discriminator network

Discriminator network, denoted as D, takes real data $(x-real)$ the corresponding conditioning information (c) as well as a fake image $(x-fake)$ and the conditioning information (c) to classify whether each input is real or fake. The discriminator network can be represented as in equation (2) and (3),

$$D(x\_real,c)\rightarrow p-real(the\ probability\ that\ x\_real\ is\ real) \tag{2}$$

$$D(x\_fake,c)\rightarrow p-fake(the\ probability\ that\ x\_fake\ is\ fake) \tag{3}$$

Step 4: Loss Function

A generator, discriminator trained using adversarial loss, which encourages generator to produce images that indistinguishable from real images given conditioning information. Loss function defined as a combination of two terms:

Generator Loss (L_G): Measures how well the generator fools the discriminator.

Discriminator Loss (L_D): Measures how well discriminator differentiates among real, fake images.

Loss function can be expressed as in equation (4) and (5),

$$L\_D=-[\log(p\_real)+\log(1-p\_fake)] \tag{4}$$

$$L\_G=-\log(p\_fake) \tag{5}$$

Step 5: Progressive training

Progressive training is the key idea in this approach. Training starts with low-resolution data’s and gradually increases the resolution. This is typically done in stages, where each stage corresponds to a higher image resolution. At each stage, a new generator and discriminator are added, and the existing generator is used as a "prior" to help train the new generator. This ensures the generation of images that are consistent with lower-resolution versions.

Step 6: Training

The networks are trained iteratively by optimizing the loss functions using gradient descent-based techniques.

Step 7: Generation

Once the model is trained, you can use the generator produce higher-quality images conditioned on given input information. The suggested model classifies legitimate,malicious transactions using PCGAN. Additionally, process anticipates incoming transactions by connecting to the blockchain smart contract.

Algorithm 4: Fake Check Scams detection Algorithm by PCGAN

```
Inputs: Balanced dataset $\lambda$
Outputs: Transactions in blockchain $K$
Initialization of dataset
Splittingof $\lambda$ into training, testing
$A_{train}$ ←Input variables from dataset
$B_{train}$ ← Target variables to dataset
$A_{train}$ ←Input variables from test dataset
$B_{train}$ ←Target variables from test dataset
Model= PCGAN $(f_e \; stimators = 100)$
Model=Model.fit $(A_{train}, A_{train})$
$G_{pred} = Model.predict(A_{test})$
Predictions=$[round(value) \; forvalue \; in \; G_{pred}$
$if \; predictions == 0 \; then$
Transaction=legitimate
$K.add(transaction)$
Else if $predictions == 1$ then
Transaction= malicious
End if
Return $K$
end
```

The suggested approach combines blockchain technology with machine learning. The underlying study makes use of bitcoin transaction database, used to train suggested machine learning model. For later use, transaction pattern kept in database studied. Transactions are carried out on Ethereum network concurrently. It believed the pattern of transactions resembles that of bitcoin transactions that are recorded in bitcoin transaction database. Furthermore, ML method taught utilizing each novel Ethereum transaction as input. Analysis, comparison of transaction pattern by bit coin transaction pattern done.

4. Result and Discussion

The experimental outcomes of suggested method are discussed in this section. This section discusses performance analysis of Block Chain with Progressive conditional Generative Adversarial Network is proposed for Detecting and Preventing Fake Check Scams (BC-PCGAN-EFDB). The proposed approach is implemented in Python version 3.6 utilizing keras, machine learning apparatus. Acquired outcomes equated with other different present models, like ML-BBEFDM [15], CCFD-BC-SAKA [16] and BC-DFCS [17], respectively.

4.1 Performance Metrics

Performance metrics comprising classification error, precision, accuracy, true positive rate, computational power, integrity, availability and Confidentiality are explored.

4.1.1 Classification error

This metric represents the overall accuracy of the detection system. It is typically measured as the percentage of misclassified instances. A lower classification error indicates a more accurate system.

4.1.2 Accuracy

This metric measures how well the system can correctly identify fake check scams. It's typically expressed as a percentage and is computed as in equation (6)

$$Accuracy = \frac{Number \; of \; correctly \; \det ected \; scams}{Total \; number \; of \; scams} \times 100 \quad (6)$$

4.1.3 False Positive Rate

This metric assesses the system's propensity to incorrectly flag legitimate transactions as scams. It's calculated as in equation (7)

$$False Positive Rate = \frac{Number of \; legitimate \; transactions \; incorrcectly \; flagged \; as \; scams}{total \; number \; of \; legitimate \; transactions} \times 100 \quad (7)$$

4.1.4 False Negative Rate

This measures the system's tendency to miss actual scams. It is computed as in equation (8),

$$False \; Negative \; Rate = \frac{\text{Number of Scams Missed by theSystem}}{\text{Total Number of Scams}} \times 100 \quad (8)$$

4.1.5 Precision

Precision quantifies the accuracy of the system's positive predictions. It's calculated as in equation (9),

$$\text{Pr}ecision = \frac{\text{Number of True Positives}}{\text{Number of true positivies+ Number of false positives}} \times 100 \quad (9)$$

4.1.6 True Positive Rate

True positive rate processes proportion of actual positive cases (i.e., real fake check scams) that the system correctly identifies as positive. It's essential to maximize this rate to minimize false negatives and catch as many scams as possible.

4.1.7 Computational Power

This metric refers to the computational resources required to run the deep learning models and maintain the blockchain. Lower computational power requirements are desirable as it reduces costs and energy consumption.

4.1.8 Integrity

Integrity ensures that the data recorded on the blockchain remains tamper-proof. Any change or manipulation of the data should be detectable and prevented, maintaining the trustworthiness of the system.

4.1.9 Availability

Availability measures how reliably the system is operational. High availability is crucial to ensure that the fraud detection and prevention system is always accessible when needed.

4.1.10 Confidentiality

Confidentiality ensures that sensitive information, such as personal and financial data, remains secure and private. Unauthorized access to this data should be prevented.

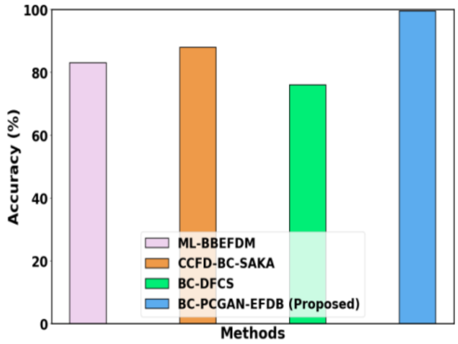

Fig 2: Accuracy analysis

The Figure 2 displays accuracy analysis. Data illustrates that method attains maximum accuracy categorizing block chain transactions malicious else real when it reaches highest peak of 0 to 100. Throughout training, accuracy stays constant after hitting the maximum value of 100. The proposed method BC-PCGAN-EFDB achieves 20.86%, 30.98% and 23.45% higher accuracy than existing methods BBEFDM, CCFD-BC-SAKA and BC-DFCS, respectively.

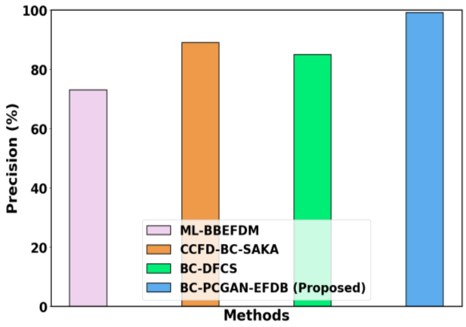

Fig 3: Precision analysis

The Figure 3 displays precision analysis. Data illustrates that method attains maximum accuracy categorizing blockchain transactions malicious else real when it reaches the highest peak of 0 to 100. Throughout the training, the precision stays constant after hitting the maximum value of 100. The proposed method BC-PCGAN-EFDB achieves 22.76%, 20.98% and 33.45% higher accuracy than existing methods BBEFDM, CCFD-BC-SAKA and BC-DFCS, respectively.

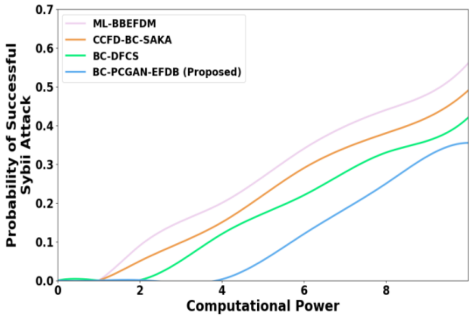

Fig 4: Computational power analysis

The Figure 4 displays computational power analysis. Proposed method BC-PCGAN-EFDB achieves 7.89%, 10.98% and 13.45% low power than existing methods BBEFDM, CCFD-BC-SAKA and BC-DFCS, respectively. Also, Figure 4 shows the Sybil attack evaluation parameters, including the number of nodes, computational capacity of attacker node, and several Sybil identities, ns = 12 and 24. The likelihood effects of various Sybil identities within the network are depicted in the provided figure. The graphic shows that chance of Sybil assault zero when there are 12 Sybil identities and no computational resources. However, as computational resources expanded from 100 by 12 Sybil identities, possibility of Sybil attack rises. It demonstrates that the probability of a Sybil assault rises as the attacker uses more computing power. Furthermore, the chance of Sybil assault zero when number of Sybil identities increased to 24 and computing resources equal to 125. On the other hand, the likelihood of an attack rises as Sybil identities' computing capabilities surpass 125. The graph shows that a high number of Sybil identities, computational resources increase possibility of Sybil attack. Results show that probability of Sybil attack influenced by quantity of Sybil identities created by attackers.

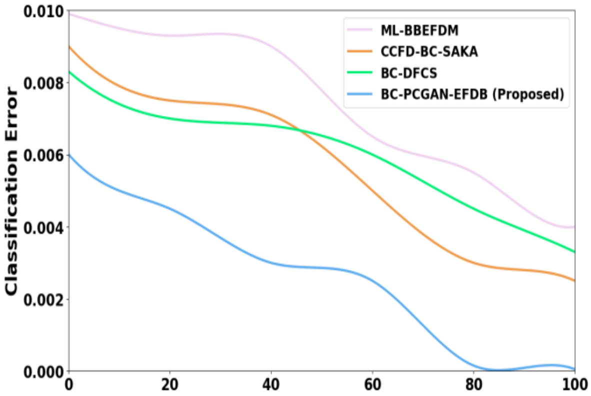

Fig 5: Classification error analysis

The error that arises during classification with BC-PCGAN-EFDB is displayed in Figure 5. It displays the error for both test and training sets of data. It is evident that as number of iterations rises, classification error falls. Training data, error is large and the figure indicates a slow decrease; for test data, the error is smaller and drops more quickly.

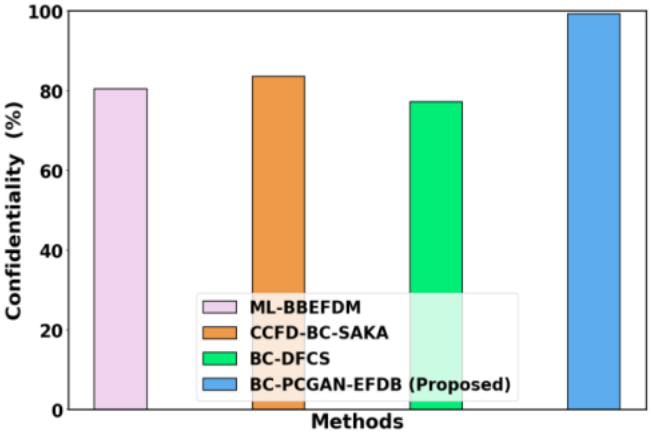

Fig 6: Confidentiality analysis

The Figure 6 displays confidentiality analysis. Proposed technique BC-PCGAN-EFDB achieves 20.86%, 30.98% and 23.45% higher confidentiality than existing methods BBEFDM, CCFD-BC-SAKA and BC-DFCS, respectively. Permissioned else private block chain, likes hyper ledger else private Ethereum networks, used to meet the requirement of confidentiality. In the scenario presented, the suggested solution is built on a permissioned blockchain network.

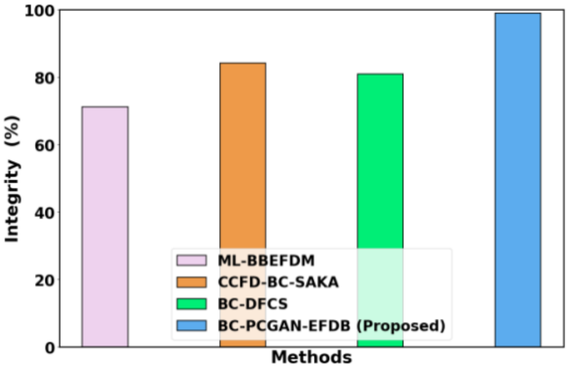

Fig 7: Integrity analysis

The Figure 7 displays integrity analysis. Proposed method BC-PCGAN-EFDB achieves 22.56%, 20.18% and 22.45% higher integrity than existing methods BBEFDM, CCFD-BC-SAKA and BC-DFCS, respectively. Data integrity is a crucial characteristic that serves to guarantee that no data alteration has occurred. Blockchain's immutability guarantees data integrity, facilitates message exchanges between parties and creates logs, events.

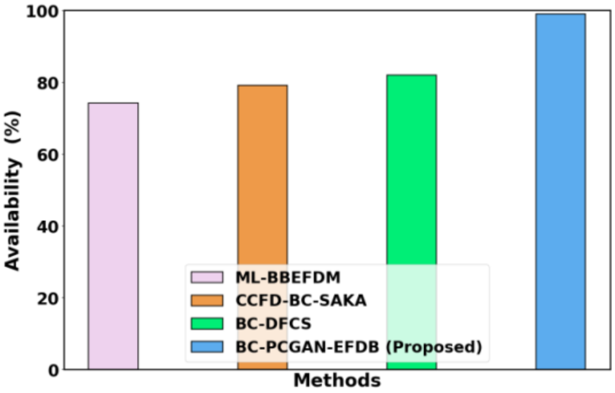

Fig 8: Availability analysis

The Figure 8 displays availability analysis. Proposed method BC-PCGAN-EFDB achieves 25.56%, 10.18% and 12.45% higher availability than existing methods BBEFDM, CCFD-BC-SAKA and BC-DFCS, respectively. It guarantees that every participant can always access deployed smart contract on blockchain. Furthermore, availability guarantees every service is constantly available. Because every transaction is kept in an Ethereum distributed ledger, defends the system from DoSassaults. As a result, there is no concern about compromise, failure, or hacking. Because it is protected by thousands of reliable mining nodes, Ethereum's ledger is highly robust to denial-of-service attacks.

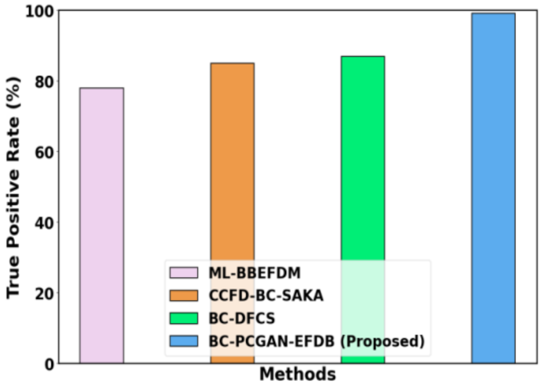

Fig 9: True positive rate analysis

The Figure 9 displays true positive rate analysis. Proposed method BC-PCGAN-EFDB achieves 25.56%, 10.18% and 12.45% higher true positive rate than existing methods BBEFDM, CCFD-BC-SAKA and BC-DFCS, respectively. It measures proportion of actual positive cases that correctly identified through system. The case of Fake Check Scams Detection and Prevention, a true positive corresponds to correctly identifying a scam or fraudulent check.

5. Discussion

Proof of Learning is a unique concept in the blockchain world. Unlike traditional Proof of Work or Proof of Stake consensus mechanisms depend on computational power or stake ownership, PoL requires participants in the blockchain network to demonstrate their knowledge or skills in a particular field. In this case, the field is likely to be related to cyber security, fraud prevention, and check authentication. The PoL blockchain would require network participants to complete learning tasks, tests, or certifications related to fraud detection and prevention. This ensures that only those with a deep understanding of the subject matter are able to participate in maintaining the blockchain. The blockchain would then record and verify these learning achievements, making it difficult for malicious actors to join the network. PCGANs are a type of generative adversarial network (GAN) that have the ability to generate data that follows a particular distribution or condition. In this context, PCGANs could be used to generate simulated datasets for training machine learning models in fraud detection. These simulated datasets would mimic a wide range of fraudulent check scenarios, allowing for better model training and validation. Handling sensitive financial data requires strong data privacy and security measures to protect users' information, Ensuring that the blockchain network can handle a large number of transactions and participants is crucial for real-world implementation, Compliance with financial and data protection regulations is vital to avoid legal issues, The effectiveness of the PoL blockchain hinges on the quality and relevance of the educational materials and testing procedures and Encouraging individuals and organizations to use this system may require incentives and awareness campaigns.

6. Conclusion

In this paper explains the proof of learning based Block Chain with Progressive conditional Generative Adversarial Network for Detecting and Preventing Fake Check Scams (BC-PCGAN-EFDB). Here, the BC-PCGAN-EFDB proposed approach is implemented and the performance metrics, like classification error, precision, accuracy, true positive rate, computational power, integrity, availability and Confidentiality are analyzed. The proposed method gives low computational power 12.56%, 15.76% and 20.98% and higher true positive rate 23.78%, 30.98% and 15.67% when comparing with existing techniques like ML-BBEFDM, CCFD-BC-SAKA and BC-DFCS, methods respectively.

Reference

[1] J. Błaszczyński, A.T. de Almeida Filho, A. Matuszyk, M. Szeląg and R. Słowiński, 2021. Auto loan fraud detection using dominance-based rough set approach versus machine learning methods. *Expert Systems with Applications*, *163*, p.113740.

[2] I. Hasan and S.A.M. Rizvi, 2022. AI-Driven Fraud Detection and Mitigation in e-Commerce Transactions. In *Proceedings of Data Analytics and Management: ICDAM 2021, Volume 1* (pp. 403-414). Springer Singapore.

[3] M. Sekar, 2022. Fraud and Anomaly Detection. In *Machine Learning for Auditors: Automating Fraud Investigations Through Artificial Intelligence* (pp. 193-202). Berkeley, CA: Apress.

[4] I. González García, and A. Mateos Caballero, 2022. A Comparison Between Bayesian Dialysis and Machine Learning to Detect Tax Fraud and Its Causes: The Case of Vat, Corporate Tax and Customs Duties in Spain. *SN Computer Science*, *4*(1), p.80.

[5] J.K. Afriyie, K. Tawiah, W.A. Pels, S. Addai-Henne, H.A. Dwamena, E.O. Owiredu, S.A. Ayeh and J. Eshun, 2023. A supervised machine learning algorithm for detecting and predicting fraud in credit card transactions. *Decision Analytics Journal*, *6*, p.100163.

[6] J.J. Bird, A. Naser and A. Lotfi, 2023. Writer-independent signature verification; Evaluation of robotic and generative adversarial attacks. *Information Sciences*, *633*, pp.170-181.

[7] W. Fang, X. Li, P. Zhou, J. Yan, D. Jiang and T. Zhou, 2021. Deep learning anti-fraud model for internet loan: where we are going. *IEEE Access*, *9*, pp.9777-9784.

[8] H.D. Nayak, L. Anvitha, A. Shetty, D.J. D'Souza and M.P. Abraham, 2021. Fraud Detection in Online Transactions Using Machine Learning Approaches—A Review. *Advances in Artificial Intelligence and Data Engineering: Select Proceedings of AIDE 2019*, pp.589-599.

[9] P. Pandey and N. Mishra, 2023. Phish-Sight: a new approach for phishing detection using dominant colors on web pages and machine learning. *International Journal of Information Security*, pp.1-11.

[10] N. Tax, K.J. de Vries, M. de Jong, N. Dosoula, B. van den Akker, J. Smith, O. Thuong, and L. Bernardi, 2021. Machine learning for fraud detection in e-Commerce: A research agenda. In *Deployable Machine Learning for Security Defense: Second International Workshop, MLHat 2021, Virtual Event, August 15, 2021, Proceedings 2* (pp. 30-54). Springer International Publishing.

[11] W. Yu, Y. Wang, L. Liu, Y. An, B. Yuan and J. Panneerselvam, 2023. A Multiperspective Fraud Detection Method for Multiparticipant E-Commerce Transactions. *IEEE Transactions on Computational Social Systems*.

[12] S. Bhardwaj and M. Dave, 2022. A Network Investigation Framework Based on Deep Learning for Fraud Transaction Detection. In *Soft Computing for Security Applications: Proceedings of ICSCS 2021* (pp. 341-349). Springer Singapore.

[13] M. Habibpour, H. Gharoun, M. Mehdipour, A. Tajally, H. Asgharnezhad, A. Shamsi, A. Khosravi and S. Nahavandi, 2023. Uncertainty-aware credit card fraud detection using deep learning. *Engineering Applications of Artificial Intelligence*, *123*, p.106248.

[14] I. Nessa, B. Zabin, K.O. Faruk, A. Rahman, K. Nahar, S. Iqbal, M.S. Hossain, M.H.K. Mehedi and A.A. Rasel, 2022, September. Recruitment Scam Detection Using Gated Recurrent Unit. In *2022 IEEE 10th Region 10 Humanitarian Technology (R10-HTC)* (pp. 445-449). IEEE.

[15] T. Ashfaq, R. Khalid, A.S. Yahaya, S. Aslam, A.T. Azar, S. Alsafari and I.A. Hameed, 2022. A Machine Learning and Blockchain Based Efficient Fraud Detection Mechanism. *Sensors*, *22*(19), p.7162

[16] P. Rani, J. Shokeen, A. Agarwal, A. Bhatghare, A. Majithia and J. Malhotra, 2022. Credit card fraud detection using blockchain and simulated annealing k-means algorithm. I *Innovative Computing and Communications: Proceedings of ICICC 2021, Volume 3* (pp. 51-59). Springer Singapore.

[17] B. Hammi, S. Zeadally, Y.C.E. Adja, M. Del Giudice and J. Nebhen, 2021. Blockchain-based solution for detecting and preventing fake check scams. *IEEE Transactions on Engineering Management*, *69*(6), pp.3710-3725.

[18] H. Wang, S. Ma, H.N. Dai, M. Imran and T. Wang, 2020. Blockchain-based data privacy management with nudge theory in open banking. *Future Generation Computer Systems*, *110*, pp.812-823.

[19] Y.J. An, P.M.S. Choi and S.H. Huang, 2021. Blockchain, cryptocurrency, and artificial intelligence in finance.In *Fintech with artificial intelligence, big data, and blockchain* (pp. 1-34). Singapore: Springer Singapore.

[20] S. Morishima, 2021. Scalable anomaly detection in blockchain using graphics processing unit. *Computers & Electrical Engineering*, *92*, p.107087.

[21] K. Kapadiya, U. Patel, R. Gupta, M.D. Alshehri, S. Tanwar, G. Sharma and P.N. Bokoro, 2022. Blockchain and AI-Empowered Healthcare Insurance Fraud Detection: An Analysis, Architecture, and Future Prospects. *IEEE Access*, *10*, pp.79606-79627.

[22] N. Altwaijry, 2023. Probability-Based Synthetic Minority Oversampling Technique. IEEE Access, 11, pp.28831-28839.

[23] H. Jia, M. Yaghini, C.A. Choquette-Choo, N. Dullerud, A. Thudi, V. Chandrasekaran, and N. Papernot, 2021, May. Proof-of-learning: Definitions and practice. In *2021 IEEE Symposium on Security and Privacy (SP)* (pp. 1039-1056). IEEE.

[24] A.A.M. Muzahid, W. Wanggen, F. Sohel, M. Bennamoun, L. Hou and H. Ullah, 2021. Progressiveconditional GAN-based augmentation for 3D objects recognition. *Neurocomputing*, *460*, pp.20-30.