

# A New Framework for Fraud Detections in Bitcoin Transactions Through Ensemble Stacking Model in Smart Cities

<sup>1</sup>Dr. M. Muni Babu, <sup>2</sup>Sai Charan Reddy, <sup>3</sup>L. Ganesh Naik, <sup>4</sup>Ranveer Singh, <sup>5T</sup>. Sai Pranay

<sup>1</sup>Associate Professor, Department of Computer Science and Engineering (AI ML), CMRIT Hyderabad, Telangana, India

<sup>2-4</sup>Student, Department of Computer Science and Engineering, (AI ML), CMRIT Hyderabad, Telangana, India

<sup>1</sup>[munibabu.m@gmail.com](mailto:munibabu.m@gmail.com)

## ABSTRACT

Bitcoin has a reputation of being used for unlawful activities, such as money laundering, dark web transactions, and payments for ransomware in the context of smart cities. Blockchain technology prevents illegal transactions, but cannot detect these transactions. Anomaly detection is a fundamental technique for recognizing potential fraud. The heuristic and signature-based approaches were the foundation of earlier detection techniques, but tragically, these methods were insufficient to explore the entire complexity of anomaly detection. Machine Learning (ML) is a promising approach to anomaly detection, as it can be trained on large datasets of known malware samples to identify patterns and features of the transactions. Researchers are focusing on determining an efficient fraud and security threat detection model that overcomes the drawbacks of the existing methods. Therefore, ensemble learning can be applied to anomaly detection in Bitcoin by combining multiple ML classifiers. In the proposed model, the ADASYN-TL (Adaptive Synthetic + Tomek Link) balancing technique is used for data balancing. Random search, grid search and Bayesian optimization are used for hyperparameter tuning. The hyperparameters have a great impact on the performance of the model. For classification, we used the stacking model by combining Decision Tree, Naive Bayes, K-Nearest Neighbors, and Random Forest. We used Shapley Additive explanation (SHAP) to interpret the predictions of the stacking model. The model also explores the performance of different classifiers using accuracy, F1-score, Area Under Curve-Receiver Operating Characteristic (AUC-ROC), precision, recall, False Positive Rate (FPR) and execution time, and ultimately selects the ideal model.

**Keywords:** Bitcoin transaction, hyperparameter tuning, machine learning, ransomware attack, stacking model, smart cities.

## I. INTRODUCTION

A decentralized and a distributed ledger that records transactions safely and publicly is called a blockchain. Each block in the chain contains a series of transactions that have been confirmed and accepted by the network. Without network consensus, a block cannot be modified or deleted once it has been added to the chain. In order to control or validate transactions, Bitcoin relies on a decentralized network rather than a centralized organization like a government or financial institution. This enables transactions that are safe, quick, and affordable without the use of middlemen like banks or payment processors. Despite these benefits, blockchain technology is not entirely secure and is still susceptible to some threats and vulnerabilities. Bitcoin has a reputation for being used for unlawful activities, due to its anonymity and lack of regulation, which attract the criminals trying to elude the authorities.

The following are some unlawful activities connected to Bitcoin and other cryptocurrencies taking place in the smart cities: Money laundering: criminals can move illegal funds undetectably across borders using Bitcoin. Dark web transactions: Bitcoin is used to pay for criminal operations including selling of guns or drugs on the dark web because of

Its anonymity. Payments for ransomware: hackers and online criminals utilize Bitcoin to pay for ransomware attacks, in which they demand money in return for access to the victim's computer or data. Bitcoin users are susceptible to hacking, which could result in financial losses and credit issues for commercial websites. Blockchain technology stops illegal activity, however, it is still vulnerable to different attacks. Thus, different strategies and how the Bitcoin ransomware attacks are performed.

Firstly, the victim receives a malicious email, then the victim downloads the email attachments. As a result, all the files in the victim's computer get encrypted. Afterward, the victim receives a message that says to send a ransom amount in the form of Bitcoin.

The victim sends a ransom amount in terms of Bitcoin and the transaction is stored in a wallet. Previous detection methods depended on heuristic and signature-based approaches too. However, these approaches were insufficient to investigate the full complexity of anomaly detection, as shown in Figure . Therefore, we needed such an efficient fraud and security threat detection model that overcomes the drawbacks of the existing models. Researchers from all around the world have been drawn to Machine Learning (ML), because heuristic approaches are used for approximate solution while ML techniques are used for accurate solution.

insufficient to investigate the full complexity of anomaly detection, as shown in Figure . Therefore, we needed such an efficient fraud and security threat detection model that overcomes the drawbacks of the existing models. Researchers from all around the world have been drawn to Machine Learning (ML), because heuristic approaches are used for approximate solution while ML techniques are used for accurate solution.

ML techniques train to learn in the same way that people do with the of continuous improvement. ML techniques can be trained on large datasets of known malware samples to recognize patterns and attributes that set them apart from benign files. ML offers a potential method for anomaly detection. This can help to detect previously unseen variants of attacks and provide real-time protection against new attacks to comprehend the advantages of ML techniques. Using ML models, we can achieve the maximum accuracy. For anomaly detection, a model comprising data collection, preparation, model development, validation, and deployment is required. Both fraud detection and security threat detection in cryptocurrency transaction

mechanisms require effective algorithms that can accurately identify fraudulent and malicious activities. However, the choice of algorithm depends on the amount of data in the dataset, and using a single algorithm may not provide sufficient accuracy. Therefore, the challenge is to develop such an effective fraud and security threat detection model that addresses the limitations of the existing algorithms. It can also handle datasets of varied sizes, and incorporate different ML techniques while exploring the influence of hyperparameters on the performance of the model. The main objective of this project is to develop an efficient, accurate, and scalable fraud detection system that can identify The suggested ML algorithms predicted how the fraud incoming transactions would behave. This is a simulation results demonstrated that the proposed model effectively detected transaction fraud. The authors addressed the limitations of traditional approaches used for detecting fraud in the

Bitcoin network in , which often rely on manual analysis and heuristics based systems. The proposed approach used trimmed K- mean as a collective anomaly detection technique that leveraged the behavior of multiple entities in the network to identify suspicious activity using ML approaches. In, the authors gave an extensive review of the approaches used to identify Bitcoin transactions associated with ransomware attacks.

This study examined how different supervised ML techniques can be used to identify Bitcoin payments made to ransomware developers. According to the findings, RF with a k-fold cross-validation method may correctly recognize new attack categories. The authors proposed a novel approach using ML for detecting malicious activities and adversarial behavior in permission-less blockchains in. neural Network (NN) can acquire large recall value and detect adversarial feature vectors. ML models are inclined towards harmful activity with the greatest number of connected accounts. RF has the highest balanced accuracy of 96.5% among supervised ML models. Singh et al. in presented a comprehensive overview of the existing approaches used for fraud detection and analysis in blockchain systems. The authors discussed the different types of anomaly detection techniques that have been proposed for blockchain systems, including rule-based systems, ML- based systems, and graph-based systems.

SVM was found to have the highest level of accuracy when employed to identify abnormalities present in the transactions taking place in the Bitcoin network. The authors described the architecture of the proposed system in which included several modules such as data collection, Generative Adversarial Network (GAN) based data generation, and ML-based entity classification for improving Bitcoin entity classification by attacking the anonymity of the Bitcoin blockchain. GAN improved Bitcoin entity classification. It also protected user privacy, and ensured transparency and fairness.

II. RELATED WORK

| Title Name                            | Problem Statement                                                                  | Solution                                                         | Method                            | Limitations                                                  |
|---------------------------------------|------------------------------------------------------------------------------------|------------------------------------------------------------------|-----------------------------------|--------------------------------------------------------------|
| ML-Based Bitcoin Fraud Detection      | Traditional rule-based systems fail to detect complex and evolving fraud patterns. | Use machine learning models to identify fraudulent transactions. | Supervised ML (SVM, RF, KNN, NB). | Cannot generalize well; single model gives limited accuracy. |
| Graph-Based Fraud Detection Model     | Difficult in capturing relationships between transactions.                         | Use graph-based models to analyze transaction networks.          | GNN, GAT, XGBoost.                | High computational cost and complexity.                      |
| Deep Learning-Based Detection         | Traditional ML struggles with large-scale and complex data.                        | Use deep learning to learn hidden patterns in transaction data.  | Neural Networks, MLP, GRU.        | Requires large training data and high resources.             |
| Ensemble Learning for Fraud Detection | Single algorithms lead to overfitting and poor performance.                        | Combine multiple classifiers to improve accuracy.                | Bagging, Boosting, Stacking.      | Increased training time and complexity.                      |

### III. METHODOLOGY

This study developed a novel ensemble-based fraud detection framework to precisely identify fraudulent Bitcoin transactions. First, the Bitcoin Heist Ransomware dataset was pre-processed by filtering transactions and changing multiclass labels into binary format to fix the class imbalance problem. Then, hybrid data balancing techniques like SMOTE-ENN and ADASYN-TL (Adaptive Synthetic + Tomek Link) were used. After balancing, several machine learning classifiers were trained, including Decision Tree (DT), Naive Bayes (NB), K-Nearest Neighbours (KNN), and Random Forest (RF). We used Random Search, Grid Search, and Bayesian Optimization to tune the hyperparameters and make the performance better. Finally, a stacking ensemble model was built. DT, NB, and KNN were used as base learners, and RF was used as a meta-learner. We used Python to build the whole system, and it worked well. evaluated using accuracy, precision, recall, F1-score, AUC-ROC, False Positive Rate (FPR), and execution time.

#### 1. SYSTEM ARCHITECTURE

The system architecture consists of several components that work together to predict adverse drug reactions caused by drug–drug interactions.

##### Remote User:

The remote user interacts with the system by submitting drug details or drug pairs to check for possible interactions and side effects.

##### Service Provider:

The service provider manages the system operations such as uploading datasets, maintaining models, and controlling user access to the platform.

##### Web Server:

The web server acts as the central communication layer. It

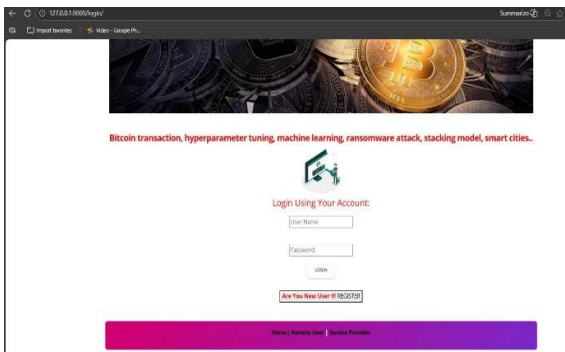


Fig 3.1 interface of system architecture

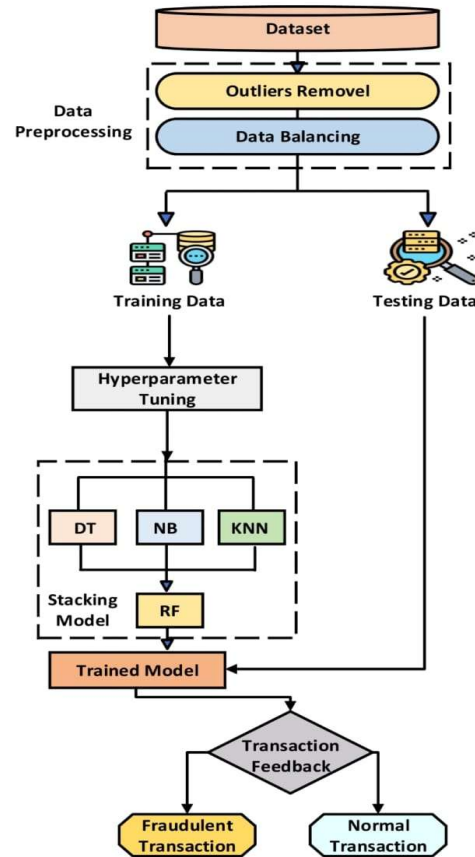


Fig 3.2 System Architecture

The suggested fraud detection system was made as a multi-stage ensemble learning framework that includes preprocessing, balancing, hyperparameter tuning, and stacking classification. The system has these main parts:

- Preparing and filtering data
- Balancing data
- Tuning hyperparameters
- Training the base learner
- Building a stacking ensemble model
- Performance assessment The architecture first loads the Bitcoin Heist dataset and filters transactions that are worth more than 0.3 BTC. After that, the dataset is turned into binary classes and made standard. Z-score normalization gets rid of outliers. Then, ADASYN-TL or SMOTE-ENN is used to balance the dataset. After balancing, each classifier's hyperparameters are adjusted. The base classifiers make predictions, which are then sent to the meta-classifier (RF) to give the final result for fraud detection. The whole system was made with Python,

and performance was evaluated using accuracy, precision, recall, F1-score, AUC-ROC, False Positive Rate (FPR), and execution time.

## 2. FLOW CHART

. The flowchart of EBD is a pictorial representation of the proposed model's algorithm, as shown in Figure 3.3. First of all, the dataset is loaded and is checked for the outliers. If outliers are found, they are removed. After that, the dataset is balanced using the balancing technique and is splitted into two sets: training and testing. Hyperparameter tuning is performed

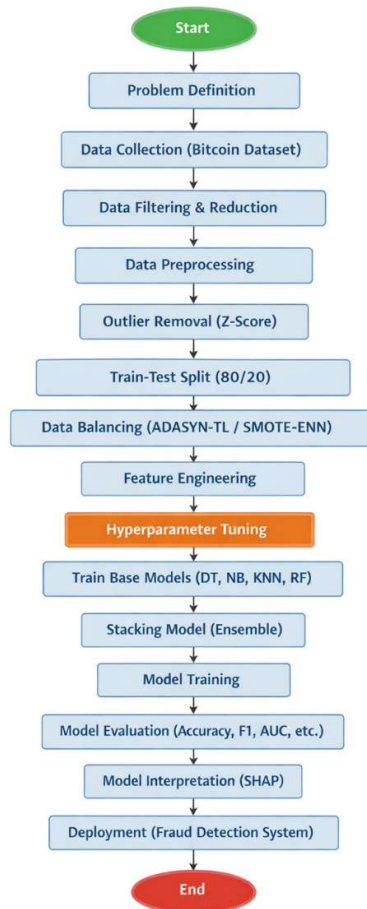


Fig 3.3 System Flowchart

Each classifier is trained on the dataset and makes prediction. The predictions of the base classifiers are added to the dataset and given to the meta-classifier as input. The meta-classifier makes the final prediction and provides accurate results, allowing users to view and save the analysis results for future reference.

## 3. MATHEMATICAL EQUATIONS

### 3.1. Dataset Preparation

We used the Bitcoin Heist Ransomware dataset for the detection of ransomware transactions. The original multiclass dataset is converted into a binary class dataset by assigning label 0 to a normal transaction and 1 to a fraudulent transaction. The dataset is publicly available on the Kaggle

It has 10 features and 2916697 instances. A lot of computing power is needed to work with such a large dataset. We don't have enough resources, so we filter out records that transfer less than 0.3 Bitcoin. This leaves us with 1387721 instances in the dataset. But the dataset is still large. We took out the data from three years (2016–2018) and now have 381464 instances left. After that, the dataset is divided into two parts: 80% and 20%. The Z-score method is also used to get rid of outliers. The dataset is very unbalanced, so data preprocessing is done on it.

### 3.2 Data Balancing Techniques

#### (ADASYN-TL and SMOTE-ENN)

The dataset is highly imbalanced, with fewer fraudulent transactions than normal transactions. To reduce bias and improve classification performance, hybrid balancing techniques were applied.

ADASYN (Adaptive Synthetic Sampling) generates synthetic minority samples as:

$$s_i = X_i + (X_{zi} - X_i)\lambda$$

where  $\lambda$  is a random value between 0 and 1,  $X_i$  and  $X_{zi}$  are minority samples.

where  $X$  represents the feature matrix,  $\bar{X}$  represents the mean, and  $n$  represents the number of samples.

### 3.3. Hyperparameter Tuning

Hyperparameter tuning was performed to optimize classifier performance.

Randomly selects hyperparameter combinations within a defined search space:

The objective is to locate an  $x$  value in  $S$  that minimizes  $f$ .  $x'$  and  $y'$  represent the global optimal solutions.

$$x' = \operatorname{argmin}_{x \in S} f(x)$$

$$y' = f(x') = \min_{x \in S} f(x).$$

It is efficient for large datasets and requires less computation compared to grid search. [3] iii. Machine Learning Classifiers K-Nearest Neighbors (KNN) Distance metric used: KNN classifies a transaction based on the majority label of its  $K$  nearest neighbors.

### 3.2. Training Configuration

The model was trained using the Ensemble Stacki with a carefully chosen learning rate schedule. To avoid overfitting, early stopping was used based on validation results. Key settings like embedding size, number of GNN layers, and dropout rate were fine-tuned through multiple experiments to get the best performance.

### 3.3. Evaluation Protocol

Model performance was evaluated using:

- Accuracy
- Precision
- Recall
- F1-score

For interaction prediction, both Top-1 accuracy and overall classification metrics were calculated on the test dataset. The evaluation ensured that the model correctly identified possible Fraud transaction and true Transactions.

## 4. ALGORITHMS

The proposed system utilizes the following core algorithms for fraud detection that happen when different transaction interact each other.

### 1. K-NEAREST NEIGHBORS (KNN):

It makes predictions by locating the k-nearest data points and figuring out which group or average value these neighbors belong to and The mathematical representation and algorithm of KNN are given as follows.

$$d(X,Y) = (X_1 - Y_1)^2 + \dots + (X_n - Y_n)^2$$

considering number of neighbors. For each data point  $X_j$  in the testing set, the algorithm calculates the distance between  $X_j$  and each data point in the training set using a distance metric.

### 2. DECISION TREE (DT) :

DT is a supervised ML technique, which is used for classification. It divides the training data into subsets depending on features and then constructs a model that resembles a tree of decisions. The mathematical representation and algorithm of DT S represents the class of data and p represents the data point. Entropy is used to find the gain. The feature with the high

$$\text{Gain}(T,X) = \text{Entropy}(T) - \text{Entropy}(T,X)$$

### 3. NAIVE BAYES(NB)

NB is a probabilistic classification technique that determines the probability of a given class using Bayes' theorem. The Bayes' theorem and its algorithm are provided below [52]. In Equation 10, A and B are events. The probability of event B depends upon the probability that event A has happened in the past

$$P(A | B) = P(B | A) \cdot P(A)$$

### 4. RANDOM FOREST (RD):

Self-Supervised Learning is applied during the RF is an ML supervised technique which is built by combining hundreds of DTs. All the DTs work in a parallel manner. RF is a bagging ensemble method. The mathematical representation and algorithm of the RF are given below

$$RF_{fii} = \text{normfii} / \text{sumT}$$

we compared different classifiers on the same data and evaluated their performance using different performance metrics. At the last step, we proposed a model trained on the same dataset and different balancing techniques and classifiers, and evaluated using different performance metrics. The proposed model is compared with their baseline classifiers in terms of accuracy, precision, recall, F1-score, and AUC-ROC.

## IV. WORKFLOW

The system workflow follows these steps:

- Dataset Loading
- Data Filtering (>0.3 BTC)
- Binary Label Conversion
- Outlier Removal (Z-score)
- Data Standardization
- Dataset Split (80:20)
- Data Balancing (ADASYN-TL / SMOTE-ENN)
- Hyperparameter Tuning
- Training Base Classifiers
- Stacking Model Training
- Performance Evaluation

This workflow ensures preprocessing and balancing are completed before classification, improving fraud detection accuracy and reducing overfitting.

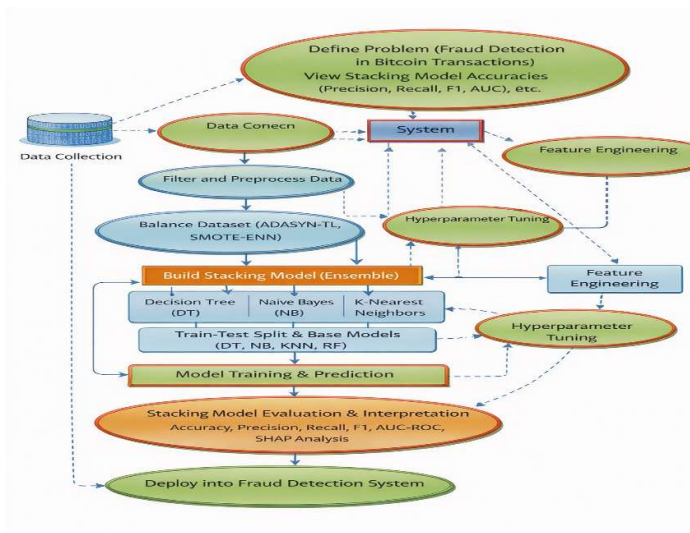


Fig 4.1 System Workflow

## V. RESULT ANALYSIS

The results demonstrate that the ADASYN-TL with the stacking model performs well among all individual classifiers 90928 in terms of accuracy and F1-score. There results how that using the stacking model it is possible to improve performance by incorporating the predictions of ML classifiers. Although the stacking model requires a lot of processing, it might be helpful for handling complex datasets. However, the stacking

The model can be expensive to run. The stacking model beats all the other benchmark classifiers in terms of accuracy (97%) and F1-score (97%). The KNN classifier works the best. This is because it works well with datasets that have non-linear decision boundaries because it doesn't make any assumptions about how the data is spread out. When it came to accuracy, precision, recall, and F1-score, RF and DT did better than NB. However, RF takes a lot longer to run than other classifiers. Also, because DT is a rule-based algorithm, even a small change in the data could change how well it works. NB doesn't work well because it assumes that all features are separate from each other.

### Transaction Dataset:

| Classifier     | Accuracy | Precision | Recall | F1-score | AUC-ROC |
|----------------|----------|-----------|--------|----------|---------|
| KNN            | 0.89     | 0.88      | 0.88   | 0.88     | 0.95    |
| DT             | 0.87     | 0.84      | 0.90   | 0.87     | 0.96    |
| NB             | 0.52     | 0.48      | 0.99   | 0.65     | 0.65    |
| RF             | 0.88     | 0.85      | 0.91   | 0.88     | 0.97    |
| Stacking Model | 0.97     | 0.96      | 0.98   | 0.97     | 0.99    |

These findings show that the proposed GNN-based framework is capable of effectively identifying potential drug– drug interactions that may lead to adverse effects.

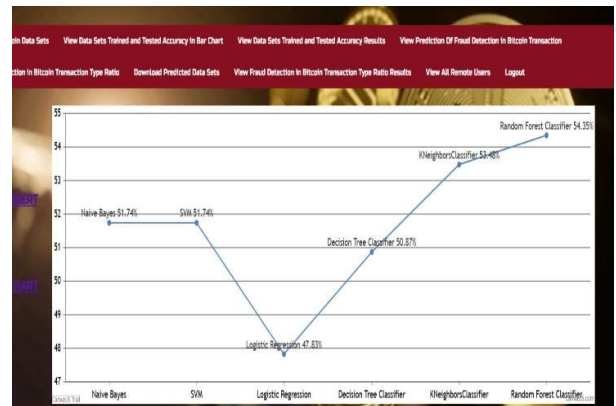


Fig 5.1 Model Accuracy Comparison Bar Chart



Fig 5.2 Performance Comparison

### TWOSIDES Dataset:

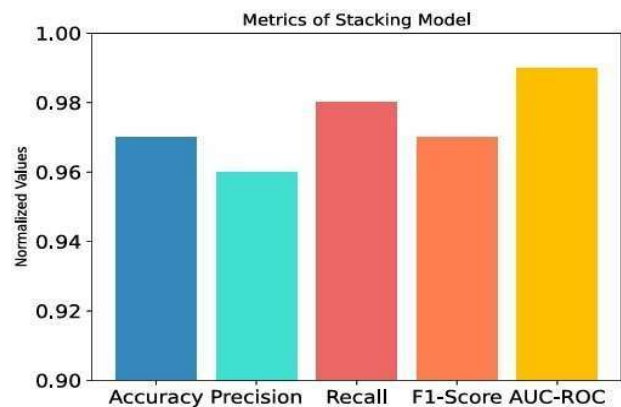
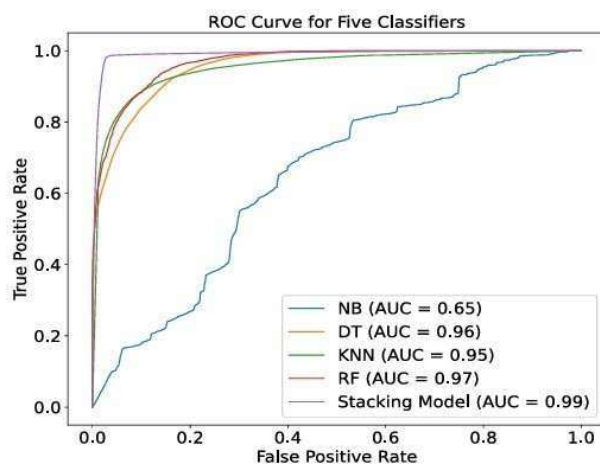


fig 5.3 Side Effect Detection Ratio

## VI. DISCUSSION

This study aims to improve fraud detection in Bitcoin transactions by using an ensemble stacking approach along with hybrid data balancing and hyperparameter optimization techniques. The results show that the proposed stacking model improves detection accuracy while reducing false positive rates, directly addressing the research objective of building a reliable and balanced fraud detection system. By combining Decision Tree, Naive Bayes, and K- Nearest Neighbors as base learners and Random Forest as a meta-learner, the model integrates the strengths of multiple classifiers instead of relying on a single algorithm. This makes the detection system more robust and less prone to overfitting compared to traditional standalone models.

## VII. CONCLUSION

This study aimed to develop an effective fraud detection model for Bitcoin transactions by addressing class imbalance, overfitting issues, and hyperparameter sensitivity. The proposed ensemble stacking framework combined hybrid balancing techniques (ADASYN-TL and SMOTE-ENN), hyperparameter tuning (Random Search, Grid Search, Bayesian Optimization), and multiple machine learning classifiers. The experimental results showed that the stacking model achieved high detection performance with 97% accuracy, 96% precision, 98% recall, 97% F1-score, and 99% AUC-ROC, while maintaining a low false positive rate of 3%. By combining multiple classifiers, the system improved generalization and reduced misclassification compared to individual models. Therefore, the research successfully achieved its objective and provided a reliable and practical solution for fraud detection in Bitcoin transactions.

## VIII. FUTURE WORK

- In future work, advanced ensemble and deep learning techniques such as XGBoost, LightGBM, and deep neural networks can be integrated to further improve detection performance.
- The system can be extended to support real-time fraud detection by integrating streaming frameworks so that suspicious transactions can be identified instantly instead of offline processing.

- Explainable AI techniques such as SHAP can be further enhanced to provide deeper interpretability, helping financial analysts understand why a transaction is classified as fraudulent.
- More advanced data balancing techniques such as Conditional GANs and Variational Autoencoders can be explored to generate more realistic fraudulent samples.
- The model can be deployed in cloud-based and distributed environments to improve scalability and practical implementation for large-scale cryptocurrency networks.

## VIII REFERENCES

- [1] N. Nayyer, N. Javaid, M. Akbar, A. Aldegheshem, N. Alrajeh, and M. Jamil, "A new framework for fraud detection in bitcoin transactions through ensemble stacking model in smart cities," *IEEE Access*, vol. 11, pp. 90916–90934, 2023. A.
- [2] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," 2008.
- [3] L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.
- [4] A. Verma and P. Singh, "*AI Driven Recruitment Automation Us*
- [5] J. R. Quinlan, "Induction of decision trees," *Machine Learning*, vol. 1, no. 1, pp. 81–106, 1986.
- [6] D. J. Hand and R. J. Bolton, "Statistical fraud detection: A review," *Statistical Science*, vol. 17, no. 3, pp. 235–255, 2002.
- [7] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in *Proc. ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining*, 2016, pp. 785–794.
- [8] J. H. Friedman, "Greedy function approximation: A gradient boosting machine," *Annals of Statistics*, vol. 29, no. 5, pp. 1189–1232, 2001.
- [9] N. V. Chawla et al., "SMOTE: Synthetic minority over-sampling technique," *J. Artificial Intelligence Research*, vol. 16, pp. 321–357, 2002.
- [10] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," 2008.

- [11] L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.
- [12] T. Cover and P. Hart, "Nearest neighbor pattern classification," *IEEE Trans. Inf. Theory*, vol. 13, no. 1, pp. 21–27, 1967.
- [13] J. R. Quinlan, "Induction of decision trees," *Machine Learning*, vol. 1, no. 1, pp. 81–106, 1986.
- [14] D. J. Hand and R. J. Bolton, "Statistical fraud detection: A review," *Statistical Science*, vol. 17, no. 3, pp. 235–255, 2002.
- [15] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in *Proc. ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining*, 2016, pp. 785–794.
- [17] J. H. Friedman, "Greedy function approximation: A gradient boosting machine," *Annals of Statistics*, vol. 29, no. 5, pp. 1189–1232, 2001.
- [18] N. V. Chawla et al., "SMOTE: Synthetic minority over-sampling technique," *J. Artificial Intelligence Research*, vol. 16, pp. 321–357, 2002.
- [19] H. He, Y. Bai, E. A. Garcia, and S. Li, "ADASYN: Adaptive synthetic sampling approach for imbalanced learning," in *Proc. IEEE Int. Joint Conf. Neural Networks*, 2008, pp. 1322–1328.
- [20] I. Tomek, "Two modifications of CNN," *IEEE Trans. Systems, Man, and Cybernetics*, vol. 6, no. 11, pp. 769–772, 1976.
- [21] D. Wolpert, "Stacked generalization," *Neural Networks*, vol. 5, no. 2, pp. 241–259, 1992.
- [22] J. Bergstra and Y. Bengio, "Random search for hyper-parameter optimization," *J. Machine Learning Research*, vol. 13, pp. 281–305, 2012. Bayesian
- [23] E. Brochu, V. Cora, and N. de Freitas, "A tutorial on optimization," *arXiv preprint Understanding Machine Learning: From Theory to Algorithms*. Cambridge University Press, 2014.