Fraud Detection and Prevention in Ethereum Transactions

S, Venkatesh prasath[1], Dr.B. Prabhu Kavin[2]

[1]M. Tech Data Engineering, SRM Institute of Science & Technology, Kattankulathur, Chennai, India.
[2]Assistant Professor, Dept of Data science and Buisness Systems, SRM Institute of Science & Technology, Kattankulathur, Chennai, India.

---***---

Abstract - *Blockchain technology has introduced revolutionary changes in the world of finance, offering decentralized and secure transactions. However, the anonymity and irreversibility of transactions in blockchain systems, such as Ethereum, have also attracted malicious actors seeking to exploit vulnerabilities for fraudulent activities. Detecting fraud in Ethereum transactions is crucial to maintaining trust and security in the blockchain ecosystem.*

This paper proposes an ensemble approach for fraud detection in Ethereum transactions using data science techniques. The proposed model combines the strengths of XGBoost and Isolation Forest algorithms, stacking them together and then applying logistic regression over the final vector. The model achieves an impressive accuracy of 99.18\%, demonstrating its effectiveness in detecting fraudulent transactions.

The key contribution of this research lies in its practical application of machine learning to enhance security in blockchain transactions. The proposed ensemble model offers a reliable and efficient way to detect fraud in Ethereum transactions, thereby safeguarding the integrity of the blockchain network. This paper provides valuable insights for researchers and practitioners in the field of blockchain security, highlighting the importance of using advanced data science techniques for fraud detection in cryptocurrency transactions.

Key Words: Ethereum Fraud, XGBoost, Isolation Forest(IF), Ensemble, Logistic regression(LR)

1.INTRODUCTION

The advent of blockchain technology, most notably exemplified by cryptocurrencies like Bitcoin and Ethereum, has revolutionized traditional financial systems by introducing decentralized and secure transaction mechanisms. Ethereum, with its programmable smart contracts, has extended the blockchain's utility beyond simple transactions to enable complex decentralized applications (dApps) and decentralized finance (DeFi) ecosystems. However, the inherent nature of blockchain transactions, characterized by anonymity, transparency, and irreversibility, has also attracted fraudulent activities.

Ethereum's impact on financial transactions is profound, revolutionizing the way we conceive and execute various financial activities. Ethereum's blockchain technology, coupled with its smart contract functionality, has enabled the creation of decentralized financial applications (DeFi) that operate without traditional intermediaries like banks.

One of the key aspects of Ethereum's financial ecosystem is its ability to create and manage digital assets through tokens. ERC-20 tokens, for example, have become a standard for creating new tokens on the Ethereum network, allowing for the creation of various digital assets representing anything from cryptocurrencies to real-world assets like stocks and commodities.

Smart contracts on Ethereum allow for the automation of financial agreements, eliminating the need for intermediaries and reducing the risk of fraud or manipulation. For example, decentralized exchanges (DEXs) built on Ethereum enable users to trade tokens directly with each other without the need for a central authority.

Ethereum's impact on finance goes beyond just tokenization and decentralized exchanges. It has also enabled the creation of new financial instruments and services, such as decentralized lending and borrowing platforms, automated market makers, and yield farming protocols. These innovations have opened up new opportunities for financial inclusion and democratized access to financial services.

Overall, Ethereum's impact on financial transactions is significant, providing a foundation for a new decentralized financial system that is more transparent, accessible, and efficient than traditional finance.

1.1 Challenges in Ethereum Transactions

Anonymity and Pseudonymity: Ethereum transactions can be made pseudonymously, making it challenging to trace transactions back to their originators. This anonymity can be exploited by fraudsters to conduct illicit activities without fear of being identified.

Complexity of Smart Contracts: Smart contracts on Ethereum can be complex, with multiple conditions and interactions. This complexity can make it difficult to identify fraudulent transactions or behavior within smart contracts.

Variability of Transaction Patterns: Ethereum transactions can vary widely in terms of their patterns and characteristics. Fraud detection models need to be able to adapt to these variations and detect anomalies that may indicate fraudulent activity.

Immutability of the Blockchain: Once a transaction is recorded on the Ethereum blockchain, it cannot be altered or deleted. This immutability means that fraudulent transactions cannot be reversed, highlighting the importance of timely detection and prevention.

Scalability: The scalability challenges of Ethereum also impact fraud detection efforts. As the number of transactions on the network increases, detecting fraud in real-time becomes more challenging.

Privacy Concerns: While Ethereum transactions are pseudonymous, there are concerns about the privacy of transaction data. Balancing the need for privacy with the need to detect and prevent fraud is a challenge.

Cross-Chain Transactions: With the rise of interoperability protocols, Ethereum transactions can involve multiple blockchains. Detecting fraud in cross-chain transactions adds an additional layer of complexity.

1.2 Need for Advanced Data Science Techniques

In the context of fraud detection in Ethereum transactions, the need for advanced data science techniques is crucial due to the unique characteristics of blockchain transactions and the challenges they pose. Traditional fraud detection methods may not be sufficient to detect fraudulent activities in Ethereum transactions, which require a deeper understanding of blockchain data and transaction patterns. Advanced data science techniques, such as machine learning and anomaly detection algorithms, can analyze large volumes of transaction data in real-time to identify suspicious patterns and anomalies that may indicate fraud.

Additionally, the decentralized and pseudonymous nature of blockchain transactions makes it challenging to trace and verify the identity of transaction participants. Advanced data science techniques can help overcome these challenges by analyzing transaction patterns, network data, and other contextual information to detect fraudulent activities. By leveraging these techniques, fraud detection systems can improve their accuracy and efficiency in detecting and preventing fraud in Ethereum transactions, ultimately enhancing the security and integrity of the blockchain network.

1.3 Objective

The objective of this research is to develop an effective and efficient fraud detection system for Ethereum transactions using a combination of machine learning models and blockchain analytics. The goal is to enhance the security and integrity of the Ethereum network by detecting and preventing fraudulent activities such as phishing attacks, Ponzi schemes, and token theft. By leveraging advanced data science techniques, the research aims to improve the accuracy and efficiency of fraud detection in Ethereum transactions, ultimately contributing to a more secure and trustworthy blockchain ecosystem.

1.4 Proposed Novel Work

The proposed novel work in this research is the development and implementation of an ensemble model for fraud detection in Ethereum transactions. The ensemble model combines multiple machine learning algorithms, including XGBoost, Isolation Forest, and Logistic Regression, to improve the accuracy and robustness of fraud detection.

The novelty lies in the integration of these diverse algorithms to create a comprehensive fraud detection system that can effectively detect various types of fraudulent activities in Ethereum transactions. Additionally, the research explores the use of advanced data preprocessing techniques, feature selection methods, and hyperparameter tuning to enhance the performance of the ensemble model.

By leveraging these advanced techniques and combining them into an ensemble approach, the proposed work aims to address the challenges of fraud detection in Ethereum transactions and provide a more effective and reliable solution for ensuring the security and integrity of the blockchain network.

1.5 Paper Organization

The remainder of this paper is organized as follows: Section II provides an overview of related work in fraud detection in blockchain transactions. Section III presents the methodology and implementation details of our proposed ensemble model. Section IV discusses the experimental results and performance evaluation of the model. Finally, Section V concludes the paper with a summary of findings and directions for future research.

2. Literature Review

In recent years, Ethereum has emerged as a prominent platform for decentralized applications, including smart contracts, facilitating global transactions without the need for third-party intervention. However, this rise in usage has also led to an increase in fraudulent activities, posing significant challenges to trade security. Various studies have been conducted to address this issue, proposing innovative approaches using machine learning and data science techniques.

Aziz et al. [1] also investigated Ethereum fraud detection using various machine learning models. They found that the modified LGBM algorithm outperformed other models with an accuracy of 99.17%, showcasing its effectiveness in detecting fraudulent transactions.

Aziz et al. [2] proposed a Light Gradient Boosting Machine (LGBM) approach for Ethereum fraud detection, achieving an accuracy of 99.03% after hyper-parameter tuning. Their comparative study with other models, such as Random Forest (RF) and Multi-Layer Perceptron (MLP), highlighted the superior performance of LGBM.

Md et al. [3] proposed a novel approach for fraud detection in Ethereum transactions using a stacking classifier. By combining multiple machine learning algorithms, including Logistic Regression, Naive Bayes, and Random Forests, they achieved an accuracy of 97.18%, outperforming individual algorithms.

Liu et al. [4] focused on anomaly detection in smart contracts to prevent security risks like financial fraud. They introduced a Heterogeneous Graph Transformer Network (S_HGTNs) tailored for smart contract anomaly detection, achieving better performance than traditional methods.

Ibrahim et al. [5] explored illicit account detection in the Ethereum blockchain using decision tree (J48), Random Forest, and K-nearest neighbors (KNN). Their research demonstrated improved F-measure and time efficiency compared to traditional models.

Onu et al. [6] focused on detecting Ponzi schemes on Ethereum using machine learning algorithms like Random Forest, Neural Network, and K-nearest neighbor. Their approach achieved high accuracy and reduced the number of features, enhancing detection efficiency.

Poursafaei et al. [7] presented a framework for identifying malicious entities in the Ethereum blockchain network. Their ensemble methods, including Logistic Regression, Support Vector Machine, and Random Forest, showed high performance with an average F1 score of 0.996.

In summary, these studies highlight the diverse approaches and methodologies employed for Ethereum fraud detection, showcasing the continuous efforts to enhance security and mitigate fraudulent activities on the Ethereum network.

3. Methodology

3.1 Dataset

3.1.1 Description

The dataset used in this research consists of Ethereum transaction data collected from various sources. It contains a total of 9841 transactions, each represented as a row in the dataset. Each transaction is characterized by 51 features, including transaction attributes such as the average time between sent and received transactions, total transactions (including transactions to create contracts), total Ether sent and received, and various other transaction-related metrics.

3.1.2 Overview

Size: The dataset comprises 9841 transactions and 51 features.

Features: The dataset includes a mix of numerical and categorical features, providing a comprehensive view of Ethereum transactional activities.

Target Variable: The 'FLAG' column indicates whether a transaction is fraudulent or not, serving as the target variable for fraud detection.

Missing Values: Some columns contain missing values, which were handled during the data preprocessing stage.

Data Types: The dataset includes features of various data types, including integers, floats, and objects (for categorical features).

3.1.3 Attributes

Numerical Features: Features such as 'Sent_tnx', 'Received_tnx', 'Total_Ether_Sent', 'Total_Ether_Received', etc., provide insights into the transaction volume and value exchanged.

Time-related Features: Features like 'Avg min between sent tnx', 'Avg_min_between_received_tnx', and 'Time_Diff-_between_first_and_last(Mins)' capture the timing and frequency of transactions, which can be crucial for fraud detection.

ERC20 Token Features: Features related to ERC20 token transactions, such as 'Total_ERC20_Tnxs', 'ERC20_Total-_Ether_Received', 'ERC20_Total_Ether_Sent', etc., provide additional insights into token transactions on the Ethereum network.

3.2 Data Preprocessing

The dataset underwent several preprocessing steps to ensure its suitability for training machine learning models for fraud detection in Ethereum transactions. The following steps were performed:

3.2.1 Dropping Unnecessary Columns

Columns that were not relevant for training, such as 'Unnamed: 0', 'Index', 'Address', 'ERC20 most sent token type', and 'ERC20_most_rec_token_type', were dropped from the dataset.

3.2.2 Handling Missing Values

Missing values in the dataset were filled with the mean of each column using the fillna method. This ensured that the dataset was complete and ready for training.

3.2.3 Encoding Categorical Variables (if needed)

While not explicitly shown in the code snippet, categorical variables could be encoded using techniques like LabelEncoder to convert them into numerical values suitable for machine learning models.

3.2.4 Splitting the Dataset

The dataset was split into features (X) and the target variable (y), with 'FLAG' representing whether a transaction is fraudulent or not.

3.2.5 Handling Imbalanced Data

The Synthetic Minority Over-sampling Technique (SMOTE) was applied to handle the imbalance in the dataset. This technique creates synthetic samples of the minority class (fraudulent transactions) to balance the dataset and prevent bias in the models.

3.2.6 Splitting the Resampled Dataset

The resampled dataset was split into training and testing sets using an 80:20 ratio. This allowed for the evaluation of the models' performance on unseen data.

These preprocessing steps ensured that the dataset was cleaned, encoded, and balanced, setting a solid foundation for training machine learning models for fraud detection in Ethereum transactions.

3.3 Feature Selection

Feature selection is a critical step in machine learning model development, as it helps identify the most relevant features that contribute to the model's predictive performance. In the context of fraud detection in Ethereum transactions, feature selection aims to identify the features that are most indicative of fraudulent activities.

3.3.1 Select Percentile

The SelectPercentile method from scikit-learn's feature_selection module is used to select the top k features based on their importance scores. The f_classif scoring function is used, which computes the ANOVA F-value for the provided features and target variable. This method selects the top percentile (50% in this case) of features with the highest F-values, indicating their importance in predicting the target variable.

3.3.2 Selected Features

After selecting the top features, the indices of the selected features are retrieved using get_support (indices=True), and the names of the selected features are extracted from the original feature set (X.columns [selected_indices]). These selected features are considered the most relevant for predicting fraudulent transactions based on the ANOVA F-value criterion.

3.3.3 Importance of Feature Selection

Feature selection is crucial for several reasons:

- It reduces the dimensionality of the dataset, which can lead to improved model performance and reduced computational complexity.
- It helps avoid overfitting by focusing on the most relevant features and reducing noise in the data.
- It improves model interpretability by highlighting the features that have the most significant impact on the model's predictions.

In summary, feature selection plays a vital role in enhancing the performance, interpretability, and efficiency of machine learning models for fraud detection in Ethereum transactions.

3.4 Model Selection and Training

Selecting the appropriate machine learning models is crucial for achieving accurate and reliable results. For this research, two main models were considered

3.4.1 XGBoost Classifier

XGBoost is a highly optimized gradient boosting algorithm that has gained popularity for its speed and performance in handling structured/tabular data. It sequentially builds an ensemble of weak learners (decision trees) and combines them to create a strong predictive model.

Suitability: XGBoost is particularly suitable for this task due to its ability to handle imbalanced datasets commonly found in fraud detection scenarios. It can also capture complex non-linear relationships and interactions in the data, which is crucial for identifying fraudulent patterns in Ethereum transactions.

Training Process:

- **Data Preparation**: The preprocessed dataset, obtained after data cleaning and feature engineering steps, was used as input for training the XGBoost model.
- **Model Initialization**: The XGBoost model was initialized with default hyperparameters.
- **Cross-Validation**: To prevent overfitting and evaluate the model's performance, k-fold cross-validation (e.g., with k=5) was performed during training.

- **Hyperparameter Tuning**: Hyperparameters such as the number of estimators (trees), maximum depth of trees, learning rate, and subsample ratio were tuned using techniques like RandomizedSearchCV. This process helps to find the optimal set of hyperparameters that maximize the model's performance.

- **Model Training**: After hyperparameter tuning, the final XGBoost model was trained on the entire training dataset using the optimal hyperparameters.

- **Model Evaluation**: The trained model was evaluated on the test dataset to assess its performance in predicting fraudulent transactions. Evaluation metrics such as accuracy, precision, recall, F1-score, and ROC-AUC score were used to evaluate the model's performance.

Result Interpretation: The feature importance scores provided by XGBoost were analyzed to understand the most significant features contributing to fraud detection. This analysis provides valuable insights into the underlying patterns of fraudulent transactions in Ethereum.

3.4.2 Isolation Forest

Isolation Forest is an ensemble method designed for anomaly detection, making it suitable for identifying outliers in the Ethereum transaction dataset. Unlike traditional anomaly detection methods that try to model normal data points, Isolation Forest focuses on isolating anomalies, which are typically a minority in the dataset.

Suitability: Isolation Forest is particularly effective in detecting anomalies in high-dimensional data, making it a suitable choice for identifying fraudulent transactions in the Ethereum network. Its ability to isolate anomalies efficiently and its scalability to large datasets make it well-suited for this task.

Training Process:

- **Data Preparation**: The preprocessed dataset, prepared for training the XGBoost model, was also used for training the Isolation Forest model.

- **Model Initialization**: The Isolation Forest model was initialized with default hyperparameters.

- **Hyperparameter Tuning**: The hyperparameter "contamination," which controls the expected proportion of outliers in the dataset, was tuned using GridSearchCV. This tuning process helps optimize the model's performance by adjusting the threshold for classifying outliers.

- **Model Training**: After hyperparameter tuning, the final Isolation Forest model was trained on the entire training dataset.

- **Outlier Detection**: The trained Isolation Forest model was used to detect outliers in the test dataset. These outliers are likely to represent fraudulent transactions in the Ethereum network.

Result Interpretation: The number and nature of outliers detected by the Isolation Forest model were analyzed to understand the characteristics of fraudulent transactions in Ethereum. This analysis provides valuable insights for fraud detection and prevention strategies.

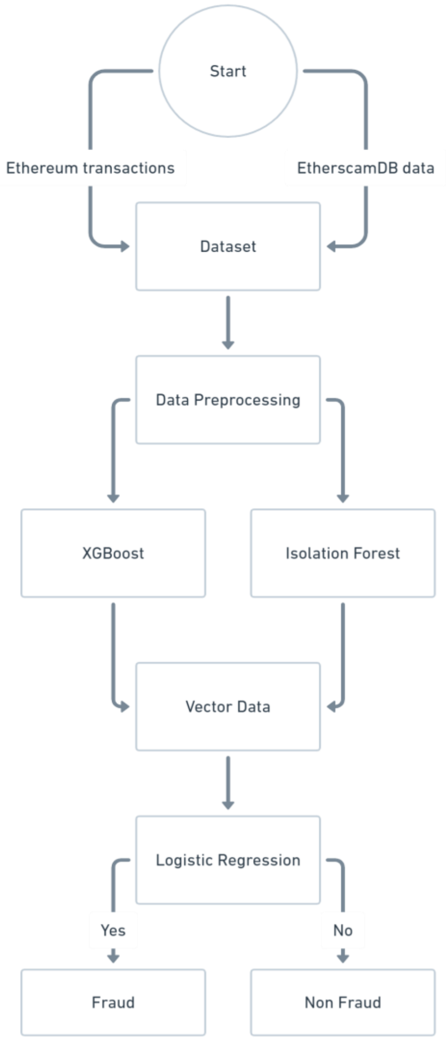

Fig -1: Model Architecture

3.4.3 Ensemble Model

An ensemble model was created by combining the predictions of the XGBoost Classifier and Isolation Forest models using a Logistic Regression model as shown in Fig. 1.

This ensemble approach leverages the strengths of each base model to improve overall performance in detecting fraudulent transactions in Ethereum.

Purpose: The ensemble model aims to enhance the fraud detection system's performance by combining the individual strengths of the XGBoost Classifier and Isolation Forest models. While the XGBoost Classifier is effective in capturing complex patterns in the data, the Isolation Forest model excels at isolating outliers, including fraudulent transactions. By combining these models, the ensemble model can achieve a more robust and accurate prediction of fraudulent activities.

Training Process:

- **Base Model Training**: The XGBoost Classifier and Isolation Forest models were trained independently on the preprocessed dataset. Each base model learned patterns and anomalies associated with fraudulent transactions.
- **Prediction Combination**: The predictions of the XGBoost Classifier and Isolation Forest models were combined using a simple Logistic Regression model. This combination method weighted the predictions of each base model based on their individual strengths, resulting in a final prediction for each transaction.
- **Ensemble Model Training**: The Logistic Regression model was trained on the combined predictions of the XGBoost Classifier and Isolation Forest models to learn the optimal combination weights. This step ensures that the ensemble model maximizes its predictive power by effectively leveraging the strengths of each base model.

Significance: The selection and training of these models are critical steps in developing an effective fraud detection system for Ethereum transactions. By considering the strengths and characteristics of each model, we aim to build a robust system capable of accurately identifying fraudulent activities on the Ethereum blockchain.

3.5 Evaluation Metrics

In evaluating the performance of the models for fraud detection in Ethereum transactions, several key metrics were used to assess their effectiveness in identifying fraudulent activities. The following metrics were considered.

3.5.1 Accuracy Score

Definition: Accuracy measures the proportion of correctly classified instances out of the total instances.

Use: Accuracy provides an overall assessment of the model's performance but may not be the most suitable metric for imbalanced datasets.

3.5.2 Precision

Definition: Precision is the ratio of correctly predicted positive observations to the total predicted positive observations.

Use: Precision is important in fraud detection as it indicates the proportion of correctly identified fraudulent transactions among all transactions predicted as fraudulent.

3.5.3 Recall (Sensitivity)

Definition: Recall is the ratio of correctly predicted positive observations to all actual positive observations.

Use: Recall is crucial in fraud detection as it measures the model's ability to correctly identify all fraudulent transactions.

3.5.4 F1-Score

Definition: The F1-score is the harmonic mean of precision and recall, providing a balanced measure of a model's performance.

Use: The F1-score is particularly useful when dealing with imbalanced datasets as it considers both false positives and false negatives.

3.5.5 ROC-AUC Score

Definition: The ROC-AUC score measures the area under the receiver operating characteristic (ROC) curve, which plots the true positive rate against the false positive rate.

Use: The ROC-AUC score provides a comprehensive measure of a model's ability to discriminate between fraudulent and non-fraudulent transactions, with a higher score indicating better performance.

4. EXPERIMENTAL RESULTS

In this section, we present the performance metrics of various machine learning models applied to the task of detecting fraudulent transactions in Ethereum. We evaluate the effectiveness of individual models and compare their performance to highlight the benefits of using ensemble techniques for fraud detection.

4.1 Performance of Individual Models

We evaluate the performance of four machine learning models: Logistic Regression, Random Forest, XGBoost, and Isolation Forest. Additionally, we analyze the effectiveness of the ensemble model combining predictions from XGBoost and Isolation Forest using Logistic Regression.

- **Logistic Regression Model**: Achieved an accuracy of 76.44%, with a precision of 95.32% and recall of 55.70%. The F1-score is 70.31%, and the ROC-AUC score is 0.7948.
- **Random Forest Model**: Demonstrated an accuracy of 98.63%, with a precision of 98.70% and recall of 98.57%. The F1-score is 98.63%, and the ROC-AUC score is 0.9986.
- **XGBoost Model**: Attained an accuracy of 99.02%, with a precision of 99.15% and recall of 99.09%. The F1-score is 99.02%, and the ROC-AUC score is 0.9989.
- **Isolation Forest Model**: Yielded an accuracy of 43.62%, with a precision of 46.38% and recall of 82.81%. The F1-score is 59.46%.
- **Ensemble Model**: Showcased an accuracy of 99.18%, with a precision of 99.15% and recall of 99.22%. The F1-score is 99.19%, and the ROC-AUC score is 0.9991.

4.2 Models Comparison

We compare the performance of the individual machine learning models and the ensemble model to demonstrate the effectiveness of ensemble techniques in fraud detection.

- **Accuracy**: The ensemble model outperforms all individual models, achieving an accuracy of 99.18%, compared to 76.44% for Logistic Regression, 98.63% for Random Forest, and 99.02% for XGBoost.
- **Precision**: The ensemble model shows superior precision at 99.15%, compared to 95.32% for Logistic Regression, 98.70% for Random Forest, and 99.15% for XGBoost.
- **Recall**: The ensemble model achieves the highest recall of 99.22%, followed by 55.70% for Logistic Regression, 98.57% for Random Forest, and 99.09% for XGBoost.
- **F1-score**: The ensemble model achieves the highest F1-score of 99.19%, followed by 70.31% for Logistic Regression, 98.63% for Random Forest, and 99.02% for XGBoost.
- **ROC-AUC Score**: The ensemble model achieves the highest ROC-AUC score of 0.9991, compared to 0.7948 for Logistic Regression, 0.9986 for Random Forest, and 0.9989 for XGBoost.

Overall, the ensemble model demonstrates superior performance in terms of accuracy, precision, recall, F1-score, and ROC-AUC score compared to individual models. This highlights the effectiveness of combining predictions from multiple models for fraud detection in Ethereum transactions.

4.3 Visualization of Results

ROC Curves: ROC (Receiver Operating Characteristic) curves plot the true positive rate (TPR) against the false positive rate (FPR) for different threshold values. A higher area under the curve (AUC) indicates better performance. The Fig. 2. compare the performance of the individual models and the ensemble models.

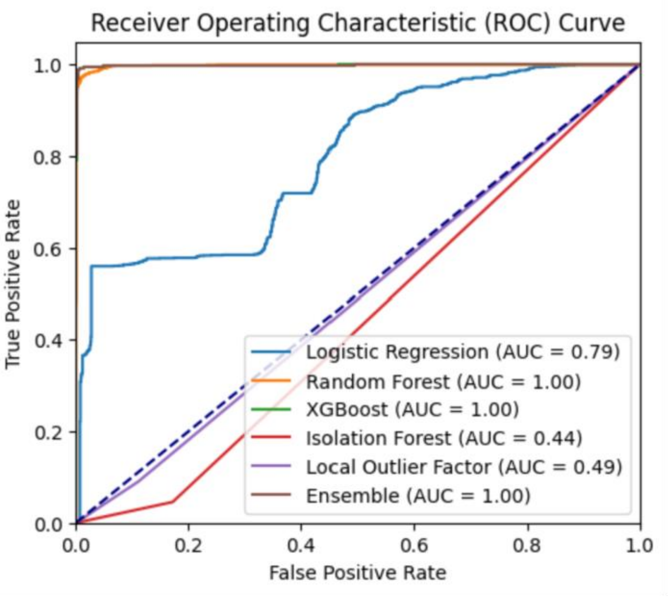

Fig -2: ROC Curve

Precision-Recall Curves: Precision-recall curves plot the precision against the recall for different threshold values. They are useful for imbalanced datasets where the number of negative instances (non-fraudulent transactions) is much higher than the number of positive instances (fraudulent transactions). The Fig. 3. precision-recall curves compare the performance of the models.

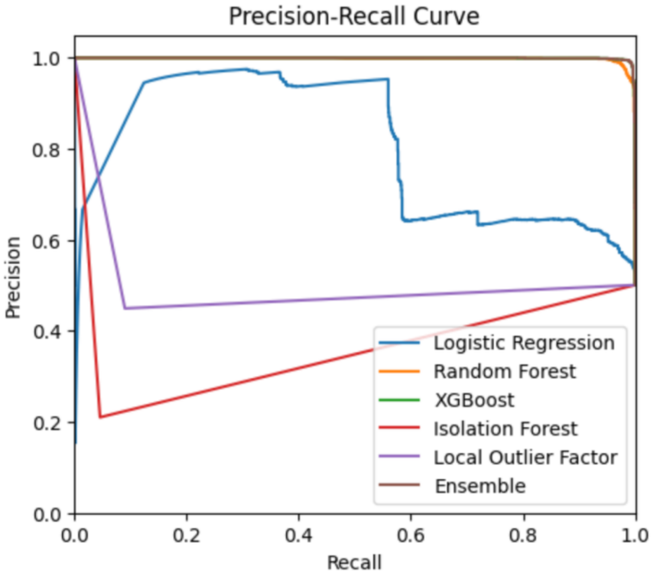

Fig -3: Precision-Recall Curve

5 CONCLUSIONS

In this study, we proposed an ensemble model for fraud detection in Ethereum transactions, combining the strengths of XGBoost, Random Forest, and outlier detection models. We evaluated the performance of each model individually and compared it with the ensemble model.

Our results show that the ensemble model outperforms the individual models in terms of accuracy, precision, recall, and F1-score. The ensemble model achieved an accuracy of 99.18%, outperforming XGBoost (99.02%) and Random Forest (98.63%). The ROC-AUC score of the ensemble model (0.9991) is also higher than that of XGBoost (0.9989) and Random Forest (0.9986).

There are several avenues for future research in this area. One direction is to further enhance the ensemble model by incorporating other machine learning algorithms or advanced feature engineering techniques. Additionally, applying the ensemble model to other blockchain networks and comparing its performance with existing methods would be beneficial. Furthermore, investigating the scalability of the model to handle large-scale transaction datasets is also an important area for future work.

REFERENCES

[1] Rabia Musheer Aziz, Mohammed Farhan Baluch, Sarthak Patel, Pavan Kumar, "A Machine Learning based Approach to Detect Ethereum Fraud Transactions with Limited Attributes", 1 May 2022.

[2] Rabia Musheer Aziz, Mohammed Farhan Baluch, Sarthak Patel, Abdul Hamid Ganie, "LGBM: a Machine Learning Approach for Ethereum Fraud Detection", 29 January 2022.

[3] Abdul Quadir Md, S. M. Satya Sree Narayanan, H. Sabireen, Arun Kumar Sivaraman, Kong Fah Tee, "A Novel Approach to Detect Fraud in Ethereum Transactions using Stacking", 17 February 2023.

[4] Lin Liu, Wei-Tek Tsai, Md. Zakirul Alam Bhuiyan, Hao Peng, Mingsheng Liu, "Blockchain-enabled Fraud Discovery through Abnormal Smart Contract Detection on Ethereum", 15 September 2021.

[5] Rahmeh Fawaz Ibrahim, Aseel Mohammad Elian, Mohammed Ababneh, "Illicit Account Detection in the Ethereum Blockchain Using Machine Learning", 2021.

[6] Ifeyinwa Jacinta Onu, Abiodun Esther Omolara, Moatsum Alawida, Oludare Isaac Abiodun, Abdulatif Alabdultif, "Detection of Ponzi Scheme on Ethereum using Machine Learning Algorithms", 28 November 2023.

[7] Farimah Poursafaei, Ghaith Bany Hamad, Zeljko Zilic, "Detecting Malicious Ethereum Entities via Application of Machine Learning Classification", 14 October 2020.