Detecting Frontrunning Attacks in Ethereum

Bachelor of Technology Thesis Project II, Spring 2024

Submitted by *Dhruv Rathi* (20EC10098)
Supervised by *Prof. Shamik Sural* & *Prof. Balaji Palanisamy*

Introduction:

1. **Blockchain's Ubiquity and Impact**: The widespread adoption of blockchain technology, catalyzed by Bitcoin's success, has transformed various sectors by offering transparency and decentralized data management.

2. **Ethereum's Significance and Leadership**: Ethereum, conceived in 2013 and launched in 2015, has advanced blockchain capabilities significantly, especially with its support for smart contracts. Surpassing Bitcoin in transaction volume, Ethereum has become the primary platform for deploying smart contracts.

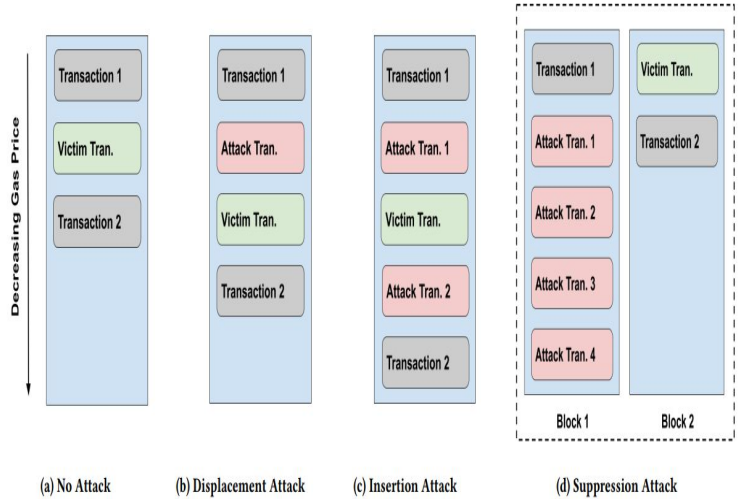
3. **Emergence of Threats and Frontrunning Attacks**: Despite Ethereum's success, it has attracted nefarious actors leading to an increase in attacks on its network. Frontrunning attacks, exploiting transaction sequencing in the pending pool, have emerged as a notable threat, facilitated by Ethereum's decentralized nature and lack of intermediaries.

Preliminaries:

1. **Ethereum Transactions:** Ethereum transactions are the fundamental building blocks of activity on the Ethereum blockchain. Each transaction represents a unit of value transfer or smart contract execution, initiated by a user or a smart contract.

2. **Transaction Ordering:** The order in which transactions are included in blocks on the Ethereum blockchain is a crucial factor in frontrunning attacks. Miners typically prioritize transactions based on the gas price, which is the transaction fee paid to the miner.

3. **Frontrunning Attacks:** Frontrunning attacks in Ethereum can be classified into three main types: displacement, insertion, and suppression. These attacks involve manipulating the order of transactions to gain an advantage or disrupt legitimate transactions.



Previous Work

In the research paper *Mitigating frontrunning attacks in ethereum* they have used only a MLP based model to classify the transactions.

In this project I have tried to use multiple complex classifiers for the same task and compare it with the MLP model.

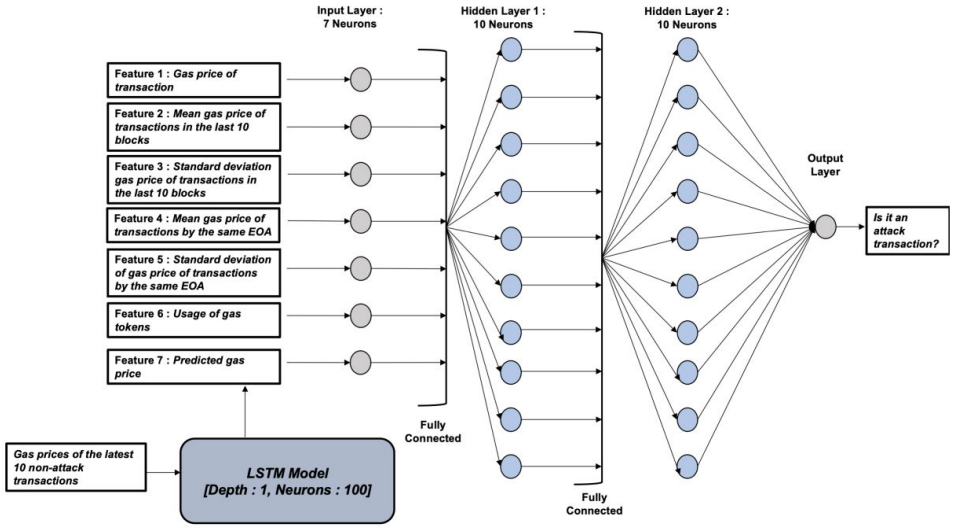


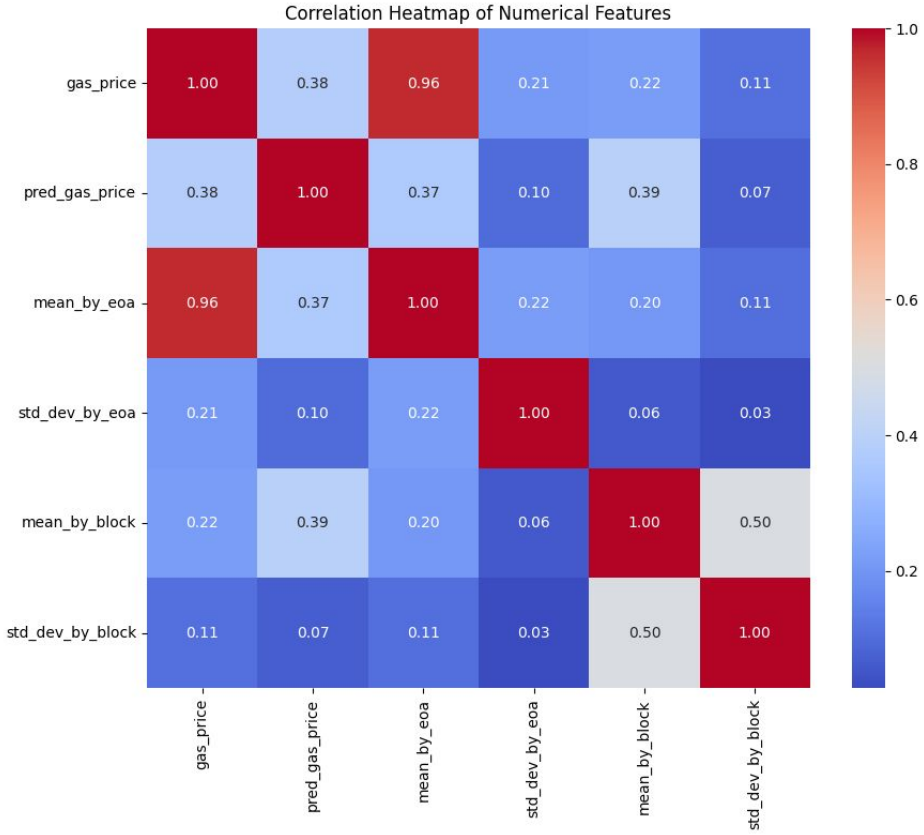
Figure 4: Multi-Layer Perceptron Model Structure

Model-Based Attack Detection:

Feature Engineering:

We identify a set of features that can be efficiently extracted from Ethereum transactions to help the machine learning model detect frontrunning attacks.

These include gas price, mean and standard deviation of gas prices in the same block and for the same externally owned account, and predicted gas price.



Model-Based Attack Detection:

Model Creation:

The two main steps for detecting frontrunning attacks in Ethereum and the machine learning models used in those steps are as follows:

1. **Prediction of Gas Price of the current transaction:** Utilized LSTM model with 100 neurons to forecast next transaction's gas price based on previous twelve transactions. Trained on latest 100 Ethereum blocks, using last fifteen transactions for prediction. Additionally, tested transformer model, yielding similar results.

2. **Classifying the transactions as Attack or Not Attack transaction**: We experiment with various machine learning models, including Multi-layer Perceptron (MLP), XGBoost, Random Forest, and Isolation Forest, to determine the most effective approach for detecting frontrunning attacks. The tree-based models, particularly XGBoost, demonstrate superior performance in terms of precision, recall, and F1-score.

Results:

TABLE 5.1: MLP Classifier Report

	Displacement(%)	Insertion(%)	Suppression(%)
Accuracy	98.12	95.44	94.40
Precision	2.34	1.15	2
Recall	1.31	1.93	8.33
f1-score	1.68	1.45	1.17

TABLE 5.2: XGBoost Classifier Report

	Displacement(%)	Insertion(%)	Suppression(%)
Accuracy	99.94	99.81	98.13
Precision	99.83	99.44	100
Recall	95.30	89.72	58.33
f1-score	97.51	94.33	73.68

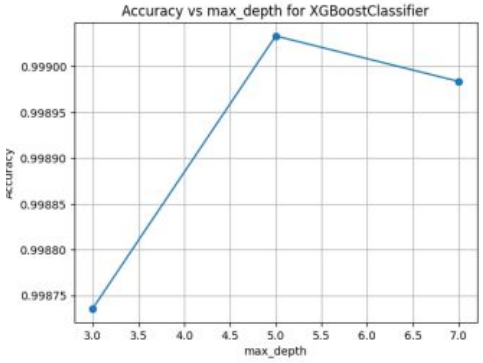
TABLE 5.3: Random Forest Classifier Report

	Displacement(%)	Insertion(%)	Suppression(%)
Accuracy	99.96	99.88	98.13
Precision	100	97.18	100
Recall	97.22	96.14	58.33
f1-score	98.59	96.66	73.68

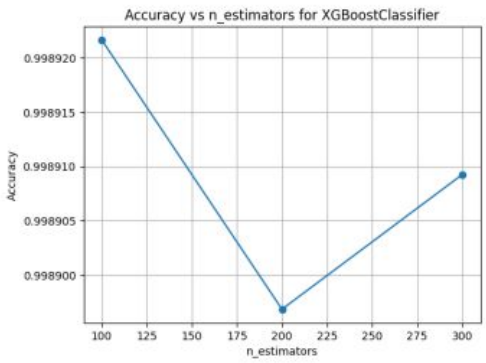
TABLE 5.4: Isolation Forest Classifier Report

	Displacement(%)	Insertion(%)	Suppression(%)
Accuracy	75.88	74.73	94.02
Precision	0.93	0.67	37.50
Recall	17.87	9.20	50
f1-score	1.78	1.24	42.85

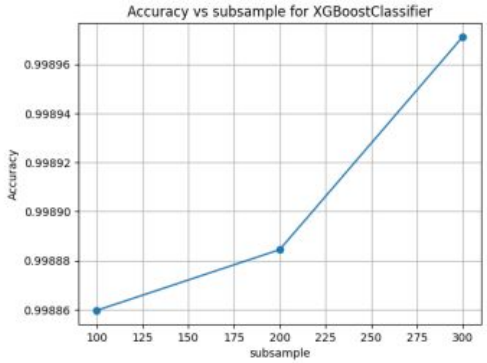
Reports of the different Classifiers used for the three types of Frontrunning Attacks.



(a) max_depth parameter

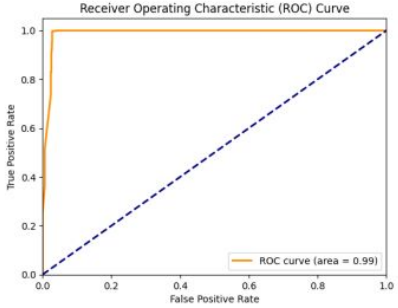


(b) n_estimators parameter

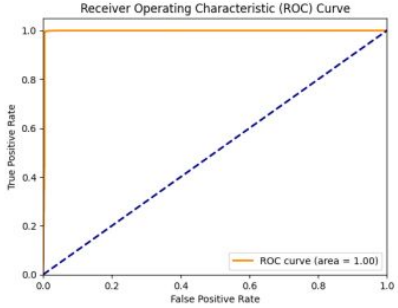


(c) subsample parameter

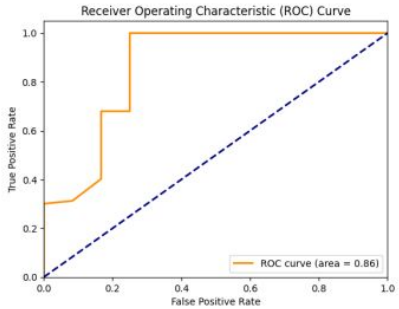
Hyperparameter tuning for XGBoost classifier



(a) Displacement Attacks



(b) Insertion Attacks



(c) Suppression Attacks

ROC curves for different Frontrunning attacks

Conclusion and Future Work:

1. **Machine-Learning Models for Frontrunning Detection**: Introduced machine-learning models to detect frontrunning attacks on Ethereum transactions. Compared MLP classifier and tree-based classifiers for identifying attack transactions, emphasizing the importance of accurate feature extraction.

2. **Feature Selection for Real-Time Detection**: Emphasized the selection of transaction features conducive to real-time detection of attacks. Chose characteristics easily obtainable from transactions within a smart contract, ensuring efficient implementation of the model directly on the blockchain.

3. **Future Integration into Smart Contracts**: Outlined the future direction of integrating classifier models into smart contracts to enable live detection and prevention of frontrunning attacks in Ethereum transactions. Aiming for proactive measures to safeguard transaction integrity on the Ethereum network.

4. **Multilabel Classification:** We can have a single multi-label classifier which could differentiate between different types of Frontrunning attacks and non-attack transactions.

Thank You!