

A hybrid machine learning approach for improved ponzi scheme detection using advanced feature engineering

Fahad Hossain¹, Mehedi Hasan Shuvo², Jia Uddin³

¹Department of Computer and Information Science, Florida International University, Miami, USA

²Department of Computer Science and Engineering, Dhaka University of Engineering and Technology (DUET), Gazipur, Bangladesh

³Department of AI and Big Data, Endicott College, Woosong University, Daejeon, South Korea

Article Info

Article history:

Received Sep 6, 2024

Revised Oct 18, 2024

Accepted Nov 19, 2024

Keywords:

Cryptocurrency fraud

Ethereum smart contracts

Feature engineering

Opcode tokenization

Ponzi scheme detection

ABSTRACT

Ponzi schemes deceive investors with promises of high returns, relying on funds from new investors to pay earlier ones, creating a misleading appearance of profitability. These schemes are inherently unsustainable, collapsing when new investments wane, leading to significant financial losses. Many researchers have focused on detecting such schemes, but challenges remain due to their evolving nature. This study proposes a novel hybrid machine-learning approach to enhance Ponzi scheme detection. Initially, we train an XGBoost classifier and extract its features. Meanwhile, we tokenize opcode sequences, train a gated recurrent unit (GRU) model on these sequences, and extract features from the GRU. By concatenating the features from the XGBoost classifier and the GRU, we train a final XGBoost model on this combined feature set. Our methodology, leveraging advanced feature engineering and hybrid modeling, achieves a detection accuracy of 96.57%. This approach demonstrates the efficacy of combining XGBoost and GRU models, along with sophisticated feature engineering, in identifying fraudulent activities in Ethereum smart contracts. The results highlight the potential of this hybrid model to offer more robust and accurate Ponzi scheme detection, addressing the limitations of previous methods.

This is an open access article under the [CC BY-SA](#) license.



Corresponding Author:

Jia Uddin

Department of AI and Big Data, Endicott College, Woosong University

Daejeon, South Korea

Email: jia.uddin@wsu.ac.kr

1. INTRODUCTION

A Ponzi scheme is a type of fraud where money from new investors is used to pay returns to earlier investors, creating an illusion of a successful investment. However, there is no actual profit being generated; the scheme depends on continuously recruiting new investors to sustain its operations. When it becomes impossible to attract new investors or when too many participants attempt to withdraw their money simultaneously, the scheme inevitably collapses, leaving many investors with significant financial losses.

Ponzi schemes deceive individuals by promising high returns with minimal risk, similar to pyramid schemes where funds from new investors are used to pay earlier participants [1]. Figure 1 illustrates the mechanics of Ponzi schemes, showing how these fraudulent operations rely on a constant influx of new investments to maintain the facade of profitability. Once the flow of new investors ceases and funds become insufficient, the scheme unravels. Critics like Roubini and Quinn argue that cryptocurrencies such as Bitcoin exhibit similar characteristics to Ponzi schemes, with early investors profiting from the influx of new participants without the generation of real value. In order to identify Smart Ponzi schemes inside the Bitcoin network, many methodologies were used, including ones focused on data mining [2]. The objective is to

utilize data mining methodologies to identify Bitcoin addresses associated with Ponzi schemes. This involves analyzing various features such as the address's lifespan, activity duration, and total value transferred to the address. The task is framed as a binary classification problem, where a classifier is trained to distinguish between addresses labeled as 'Ponzi' and those labeled as 'non-Ponzi'. Survival analysis has been used in the realm of Bitcoin to identify the elements that contribute to the persistence of scams [3]. Vasek and Moore discovered that increasing interaction between fraudsters and their victims prolongs the lifespan of scams, whereas schemes tend to have shorter durations when crooks create their accounts on the same day they initiate the scam. Additional research has provided more detailed information on these fraudulent schemes by collecting and examining documented instances of scams [3], [4]. The use of smart contracts has significantly enabled the spread of Ponzi schemes. By using smart contracts, scheme initiators may operate anonymously without any need to reveal the contract's name or the cash withdrawn from it once it is established. Moreover, the inherent decentralization of smart contracts implies that once they are activated, there is no mechanism in place to terminate them or provide compensation to those who have suffered financial losses.

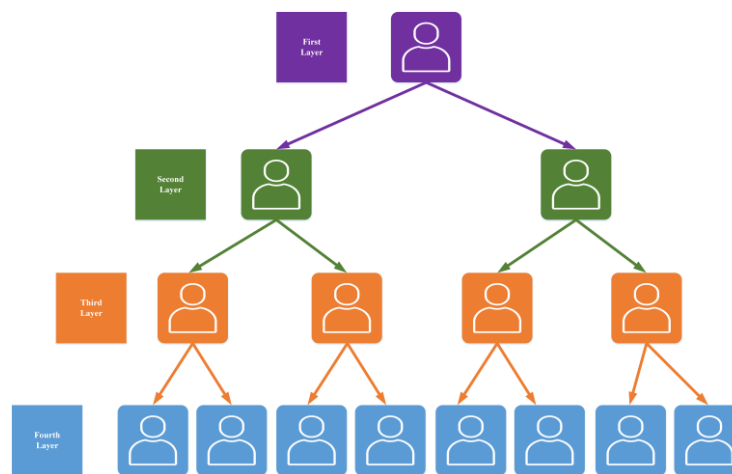


Figure 1. Ponzi scheme contract categorization

Furthermore, Ponzi schemes are increasingly using Bitcoin as a payment system alongside conventional criminal endeavors like ransomware [5]-[7] and money laundering [8], [9]. These scams present themselves as high-yield investment programs, but in reality, they only reimburse investors with funds contributed by new members. Consequently, these scams collapse when they fail to draw in new investors [10]. Currently, examining Bitcoin frauds typically necessitates an initial phase that demands significant effort and time, involving manual or partially automated online searches [11]-[14] to gather Bitcoin addresses associated with the fraud. Automated examination of the scam's effect can only occur after this step, namely by evaluating blockchain transaction inspections. However, these methods fall short when fraudulent locations remain hidden, such as when they are only accessible through the deep or dark web or privately shared with authorized individuals. In such situations, using technologies that can independently search the Bitcoin blockchain for suspicious behaviors and detect addresses associated with fraudulent conduct would be quite beneficial. The core tenets of a Smart Ponzi scheme are: participants are required to make a minimum investment in order to join the plan, payments to investors will only begin if there are enough amounts of cash available, the strategy fails when it no longer attracts new investors, and insufficient cash to compensate investors will result in the scheme's failure.

This paper introduces a novel semantic-aware method for the detection of Ponzi schemes through the use of advanced feature engineering. In order to improve the accuracy of Ponzi scheme detection, we implement a two-pronged approach. Initially, an XGBoost classifier is utilized to train on structured features extracted from financial transaction data. Subsequently, a tokenizer is used to encode opcode sequences extracted from smart contracts, training a GRU on these sequences to capture temporal patterns. Features derived from both the XGBoost classifier and GRU [15] are concatenated to form a comprehensive feature set. Finally, a final XGBoost model is trained on these concatenated features to leverage both the structured financial data and temporal patterns encoded in opcode sequences. The performance of the final model is evaluated rigorously to assess its effectiveness in detecting Ponzi schemes with improved accuracy and reliability.

The contributions of our paper are: (i) have proposed and implemented a novel hybrid model for recognizing smart Ponzi schemes in Ethereum contracts, (ii) have engineered new features that will enhance the performance of the model, (iii) constructed a GRU on opcode sequences in order to extract temporal features, and (iv) the proposed model mitigates the false positive rate as well as the false negative rate which is promising enough compared to the existing system in our context.

Our evaluation is driven by three research questions that address the key issue of whether and how the notion of detecting Ponzi schemes.

- RQ1: How can ML models be used to detect Ponzi schemes?
- RQ2: How effective are hybrid machine learning approaches combining structured financial data with smart contract analysis?
- RQ3: How can the accuracy of Ponzi scheme detection models be evaluated?

2. RELATED WORK

With the advancement of blockchain technology new variants of the Ponzi scheme also emerged. In 2018, it was found that there are around four hundred Ponzi schemes in Ethereum by Chen *et al.* [16] and they extracted the features from operation codes by using machine learning and data mining. Wang and Huang utilized the n-gram algorithm for enhanced opcode feature extraction and integrated it with contract account features [17]. They also introduced adaptive synthetic sampling (ADASYN) to handle class imbalance in the data and used the improved AdaBoost classifier for identifying Ponzi scheme contracts. In 2022, Aljofey *et al.* [18] tackled the problem of detecting smart Ponzi contracts over the Ethereum blockchain by constructing an effective detection model using data mining techniques. The process they used included expanding the dataset of smart Ponzi contracts, balancing the data with adaptive synthetic sampling, and creating four different feature sets drawn from the operation codes (opcodes) of smart contracts. These specific features, such as opcode frequency, count vector, n-gram term frequency-inverse document frequency (TF-IDF), and opcode sequence attributes, strengthened the model's dependability after the smart contract's introduction to the Ethereum Blockchain.

Aljofey *et al.* [19] provides important insights into the understanding of Ponzi scheme detection in Ethereum, highlighting the efficacy of ensemble models that utilize opcode-based features. Xu *et al.* [20] dived deep into the challenge of detecting Bitcoin mixing services, which boost anonymity by unclear fund flow but are often exploited for illegal activities like money laundering. Ibba *et al.* [21], Ethereum's capabilities for peer-to-peer programming and smart contract publishing are explored, focusing on the detection of Ponzi schemes using machine learning. Furthermore, Yu *et al.* [22] proposed a GCN model (graph convolutional network) to identify Ponzi contracts within Ethereum. The study showcases that the proposed GCN-based model offers promising results compared to general machine learning methods, contributing to the ongoing efforts to maintain the sustainable development and security of the Ethereum platform. Bartoletti *et al.* [2] show a data mining approach for detecting Bitcoin addresses associated with Ponzi schemes. They leverage the pseudonymity of Bitcoin to trace fraudulent investments that rely on recruiting new users to repay existing ones. Zhang *et al.* [23] highlights two existing challenges in detecting such schemes in the blockchain: incomplete features for detection and inefficient algorithms. The authors propose an innovative approach that combines bytecode features with user transaction and opcode frequencies, creating more comprehensive features. Chen *et al.* [16] propose SADPonzi, an innovative semantic-aware detection approach where the model utilizes heuristic-guided symbolic execution to generate semantic information for feasible paths in smart contracts, identifying investor-related transfer behaviours and distribution strategies.

Fan *et al.* [24] propose a novel detection method for Ponzi schemes on smart contract platforms. The approach utilizes ordered target statistics (TS) to process category features, employs data augmentation to address dataset imbalance, and adopts the ordered boosting algorithm to combat prediction shifts. Lou *et al.* [25] proposed an improved CNN model for Ponzi scheme detection. Zheng *et al.* [26] presented a novel method that uses a large dataset to extract features from the perspectives of bytecode, semantics, and developers to address these difficulties. They demonstrate higher accuracy in identifying clever Ponzi schemes, even at the beginning of their formation, using a machine learning-based model dubbed the multi-view cascade ensemble. Zhang *et al.* [27], a PD-SECR detection approach is presented which uses the SMOTEENN-mixed sampling algorithm to improve the combined model of convolutional neural networks and random forests. However, these studies are not without their limitations, despite their contributions. Some of the datasets they researched were small, for example, the proposed SADPonzi model proposed by Chen *et al.* [16] experimented with only 1395 samples. Also, some of the proposed models did not provide a clear picture of false positive and false negative rates [26]. Liang *et al.* [28] proposed a PonziGuard Ponzi scheme using a contract runtime behavior graph (CRBG). The limitations of CRBG are computational

complexity, scalability, operate only on a certain level of abstraction. Onu *et al.* [29] applied several machine learning algorithms for Ponzi detection to address the negative impact of Ponzi schemes using the Ethereum transactions dataset. The size of the dataset used to validate the models is small in size.

3. PROPOSED METHOD

Figure 2 especially Figure 2(a), we illustrate the steps involved in developing and evaluating our hybrid machine-learning approach for Ponzi scheme detection. The methodology consists of several crucial stages, including dataset selection, data preprocessing, feature extraction, and model evaluation.

3.1. Dataset

We collected the dataset from the Kaggle website, which provides detailed information on Ethereum smart contracts [30]. The dataset comprises 3786 entries and includes four key features: address, opcode, label, and creator. The address feature lists the unique identifiers for each smart contract, while the opcode feature contains the disassembled bytecode instructions of these contracts. The label feature indicates whether a smart contract is a Ponzi scheme, as determined through manual inspection. Finally, the creator feature identifies the individuals or entities that created these smart contracts. This dataset serves as the foundation for our analysis and model training.

3.2. Data preprocessing

Effective data preprocessing is essential for building a robust machine-learning model. This study's preprocessing steps include data cleaning, label encoding, and scaling the data using the MinMax Scaler which is illustrated in Figure 2(b).

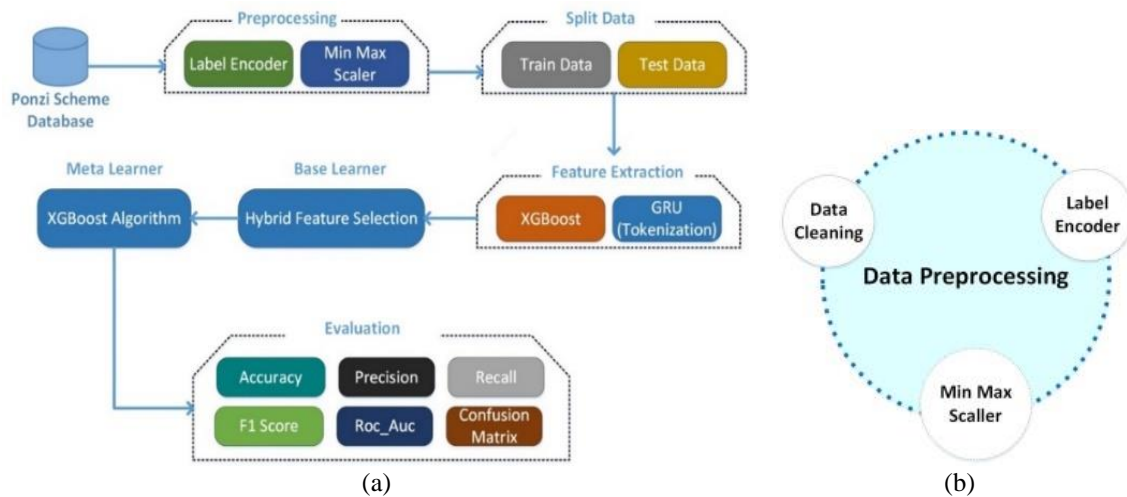


Figure 2. Block diagram of (a) proposed methodology and (b) data preprocessing

3.2.1. Data cleaning

Data cleaning is a crucial step in preprocessing to ensure the quality of the data. This involves addressing missing values by either ignoring them or filling them in with appropriate estimates. Additionally, noisy data, which may result from random errors or variances, is smoothed using techniques like binning, regression, and clustering. For example, binning organizes data into equal-sized bins, allowing for the replacement of values with the bin's mean or median. Outliers, or data points that deviate significantly from others, are identified and removed using clustering methods, where inconsistent data is separated from the main groups.

3.2.2. Label encoder

Label encoding [31] is used to convert categorical labels into numerical values, making them suitable for machine learning algorithms. This process assigns a unique integer to each category, transforming the dataset into a format that the model can easily interpret. For instance, if y is the categorical variable, the label encoder maps each category y_i to a numerical value \hat{y}_i as:

$$\hat{y}^i = \text{LabelEncoder}(y^i)$$

this transformation is essential when dealing with categorical data in the training process.

3.2.3. MinMax scaler

The MinMax scaler is applied to normalize the data by scaling all the feature values to a specific range, typically between 0 and 1. This ensures that no feature dominates the learning process due to its magnitude. The scaling is done using the formula:

$$\hat{x} = \frac{x - x_{\min}}{x_{\max} - x_{\min}}$$

where x represents the original feature value, and x_{\min} and x_{\max} are the minimum and maximum values of that feature, respectively. Normalizing the data in this way helps improve the performance and convergence speed of the machine learning model. After preprocessing of the dataset, we split it into training and testing sets. A classic 80-20 divide was utilized to make sure the model had enough material to learn from while still having sufficient data left for an unbiased evaluation.

3.3. Feature extraction/base learner

Feature extraction/ Base Learner is a crucial step in the proposed hybrid approach, as it enhances the model's ability to detect Ponzi schemes by leveraging the strengths of both static and sequential data analysis. In this stage, we use two different models-XGBoost and gated recurrent unit (GRU)-to extract meaningful features from the dataset.

3.3.1. XGBoost classifier

The first step in feature extraction involves training an XGBoost classifier on the dataset [32]. XGBoost is a powerful gradient-boosting algorithm known for its efficiency and high performance in classification tasks. Once the model is trained, we extract the most important features identified by the XGBoost classifier. These features capture the static relationships within the data and serve as an essential input to our hybrid approach. Mathematically, XGBoost minimizes the following objective function during training:

$$\text{Obj}(\theta) = \sum_{i=1}^n L(y_i, \hat{y}_i) + \sum_{k=1}^K \Omega(f_k)$$

where $L(y_i, \hat{y}_i)$ is the loss function, and $\Omega(f_k)$ is the regularization term to prevent overfitting. The important features extracted from XGBoost are those that contribute most significantly to minimizing this objective function, thus providing valuable insights into the dataset.

3.3.2. Gated recurrent unit

In parallel, we employ a GRU model [33] to analyze the opcode sequences in the dataset. GRUs, a variant of recurrent neural networks (RNNs) [34], are well-suited for handling sequential data, such as opcode sequences found in smart contracts. The GRU model learns patterns over time and extracts features that reflect the temporal dependencies in the data. The GRU cell's operations can be described mathematically as follows:

$$\begin{aligned} z_t &= \sigma(W_z \cdot [h_{t-1}, x_t]) \\ r_t &= \sigma(W_r \cdot [h_{t-1}, x_t]) \\ \tilde{h}_t &= \tanh(W \cdot [r_t * h_{t-1}, x_t]) \\ h_t &= (1 - z_t) * h_{t-1} + z_t * \tilde{h}_t \end{aligned}$$

where z_t is the update gate, r_t is the reset gate, h_t is the hidden state, and x_t is the input at time step t . The GRU model generates features that capture the sequential dynamics of the data, which are crucial for understanding complex patterns in smart contract behavior.

3.3.3. Hybrid feature selection

After extracting features from both the XGBoost classifier and the GRU model, we concatenate these features to create a comprehensive hybrid feature set. This combined feature set leverages the strengths of both models, integrating static and sequential information. The hybrid feature selection allows the subsequent meta-learner to make more informed predictions, improving the overall detection accuracy.

3.4. Meta-learner (XGBoost algorithm)

The meta-learner in our approach utilizes the XGBoost algorithm to combine and refine features extracted from the base learners. This section elaborates on the role of XGBoost as a meta-learner, and its integration into the overall methodology.

3.4.1. XGBoost as a meta-learner

In our hybrid machine-learning framework, the XGBoost algorithm is employed as a meta-learner to leverage the features extracted from the XGBoost classifier and the GRU model. The role of the meta-learner is to combine these features and make final predictions by integrating the insights gained from both base models. XGBoost, known for its high performance and accuracy in classification tasks, enhances the predictive power of our approach by effectively handling complex interactions between features.

3.4.2. Feature combination and training

The combined feature set, consisting of features from both the XGBoost classifier and the GRU model, is used as input to the XGBoost meta-learner. This integration allows the meta-learner to capture and exploit the complementary strengths of the base models. Training the XGBoost meta-learner involves fitting the model on this hybrid feature set, enabling it to make informed decisions based on the combined insights. The algorithm's gradient boosting framework further refines the model's predictions by minimizing the error through iterative boosting.

4. RESULTS AND DISCUSSION

In this section, we present the results of our hybrid machine-learning approach for Ponzi scheme detection, focusing on key performance metrics and visual representations. The performance of our model is evaluated using several key metrics: accuracy, precision, recall, and F1-score which is illustrated in Figure 3.

Accuracy reflects the overall performance of the model by calculating the ratio of correctly predicted instances to the total instances. Precision indicates the percentage of true positive predictions among all positive predictions generated by the model. Recall assesses the model's ability to correctly identify actual positive instances. The F1-score, which is the harmonic mean of precision and recall, offers a balanced measure that accounts for both metrics. For our model, the accuracy is 96.83%, with a precision of 78.38%, a recall of 64.45%, and an F1-score of 70.73%.

The confusion matrix provides a detailed breakdown of the model's classification performance by comparing predicted labels to the actual labels (Figure 4). It is an essential tool for understanding the types of errors made by the model and assessing its effectiveness. Figure 4(a) illustrated the confusion matrix for the hybrid machine-learning model. Here's what each value in the confusion matrix represents: true negatives (TN) = 705: The number of Non-Ponzi instances correctly classified as Non-Ponzi, false positives (FP) = 8: the number of Non-Ponzi instances incorrectly classified as Ponzi, false negatives (FN) = 16: the number of Ponzi instances incorrectly classified as Non-Ponzi, and true positives (TP) = 29: the number of Ponzi instances correctly classified as Ponzi. By analyzing the confusion matrix, we can see that the model performs well in identifying Non-Ponzi instances, with a high number of TN. However, there is a trade-off between FP and FN, which reflects areas where the model could be improved.

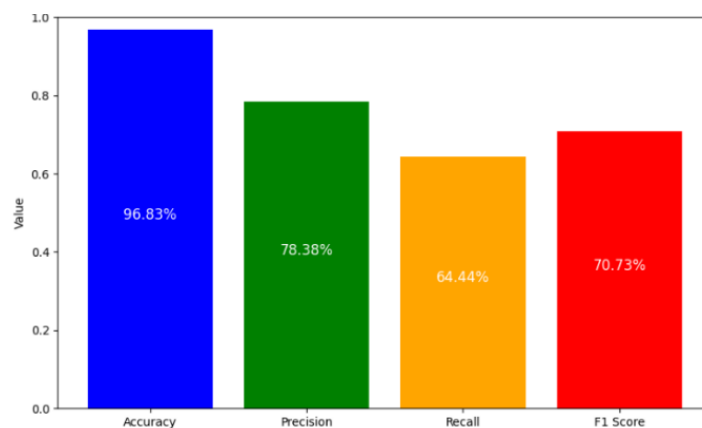


Figure 3. Performance metrics of accuracy, precision, recall, and F1-score

Understanding these metrics helps us refine our approach to achieve better performance and accuracy in detecting Ponzi schemes. The receiver operating characteristic (ROC) curve illustrates the model's performance across different thresholds. The area under the curve (AUC) is a key indicator of the model's ability to discriminate between Ponzi and Non-Ponzi instances. Our model achieves an AUC of 0.93, indicating a high level of performance. Figure 4(b) displays the ROC curve for our model.

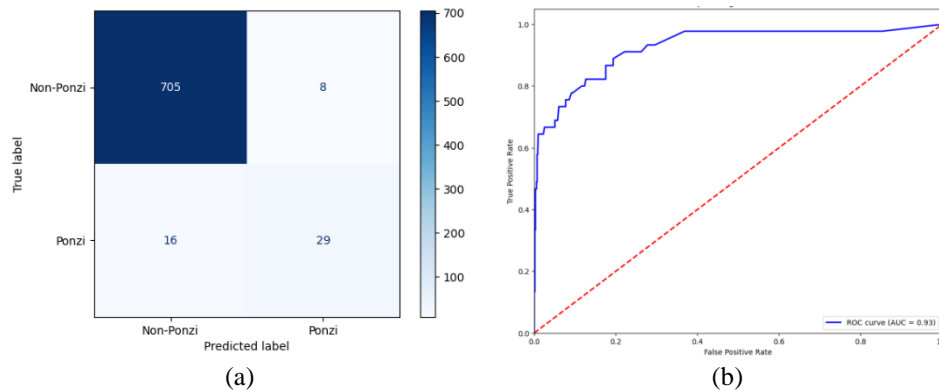


Figure 4. Performance of Ponzi (a) confusion matrix and (b) ROC curve

The results of our hybrid machine learning approach show strong performance in detecting Ponzi schemes. The model achieved an accuracy of 96.83%, demonstrating its overall reliability. Precision was 78.38%, indicating a good proportion of correctly identified Ponzi schemes among those predicted. Recall was 64.45%, reflecting the model's ability to correctly identify Ponzi schemes from all actual instances. The F1-score of 70.73% balances these two metrics, confirming the model's effectiveness. Additionally, the ROC curve with an AUC of 0.93 further highlights the model's strong discriminatory power between Ponzi and Non-Ponzi instances. Overall, these results validate the robustness of our approach in accurately identifying fraudulent activities in Ethereum smart contracts. In Table 1 we have discussed the research question and answer for explaining our methods.

Table 1. Research question and answer based on our research

SL	Question	Answer
RQ1	What role does XGBoost play in the proposed methodology?	XGBoost is initially used as a classifier to identify patterns and extract features from the dataset, which are then used in conjunction with GRU-extracted features to enhance the final model's predictive accuracy.
RQ2	How does the GRU model contribute to the detection process?	The GRU model processes and tokenizes opcode sequences from Ethereum smart contracts, capturing sequential dependencies and extracting relevant features, which are then combined with XGBoost features for improved detection.
RQ3	How does advanced feature engineering improve Ponzi scheme detection?	Advanced feature engineering enhances detection by extracting and combining relevant features from different models, allowing the hybrid model to better capture the characteristics of Ponzi schemes, leading to improved predictive accuracy.
RQ4	What is the potential impact of this research on the broader field of fraud detection?	This research has the potential to significantly improve fraud detection in blockchain environments, offering a robust, accurate, and scalable solution that can be adapted to various types of financial fraud beyond Ponzi schemes.

5. CONCLUSION

Ponzi schemes are deceptive financial operations that promise high returns with little risk, often leading to significant financial losses when they collapse. Detecting such schemes, especially within the complex and evolving landscape of blockchain technology, remains a significant challenge. In response to this challenge, our research introduced a hybrid machine learning approach that combines the strengths of XGBoost and GRU models, coupled with advanced feature engineering, to improve the detection of Ponzi schemes in Ethereum smart contracts. Our methodology began by leveraging XGBoost to identify initial patterns and extract relevant features. Simultaneously, we used a GRU model to process opcode sequences from smart contracts, extracting sequential features that capture the intricacies of transaction patterns. By integrating the features from both models, we trained a final XGBoost classifier that demonstrated superior performance compared to traditional methods. The hybrid model achieved an impressive detection

accuracy of 96.83%, along with strong precision, recall, and F1-score metrics, showcasing its robustness in identifying fraudulent activities. The results of our study highlight the effectiveness of combining machine learning models with sophisticated feature engineering to tackle the complexities of Ponzi scheme detection. This approach not only addresses the limitations of previous methods but also sets a new standard for accuracy and reliability in fraud detection within blockchain environments. Future work could further refine this methodology, exploring additional models and expanding its application to other forms of financial fraud, potentially broadening the impact of our research on the broader field of fraud detection.

ACKNOWLEDGEMENTS

This research was funded by Woosong University Academic Research 2024.





REFERENCES

- [1] M. Artzrouni, "The mathematics of Ponzi schemes," *Mathematical Social Sciences*, vol. 58, no. 2, pp. 190–201, 2009, doi: 10.1016/j.mathsocsci.2009.05.003.
- [2] M. Bartoletti, B. Pes, and S. Serusi, "Data mining for detecting bitcoin ponzi schemes," in *2018 Crypto Valley Conference on Blockchain Technology (CVCBT)*, Jun. 2018, pp. 75–84, doi: 10.1109/CVCBT.2018.00014.
- [3] M. Vasek and T. Moore, "Analyzing the bitcoin ponzi scheme ecosystem," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 10958 LNCS, 2019, pp. 101–112.
- [4] Y. Boshmaf, C. Elvitigala, H. Al Jawaheri, P. Wijesekera, and M. Al Sabah, "Investigating MMM ponzi scheme on bitcoin," in *Proceedings of the 15th ACM Asia Conference on Computer and Communications Security*, Oct. 2020, pp. 519–530, doi: 10.1145/3320269.3384719.
- [5] M. Spagnuolo, F. Maggi, and S. Zanero, "Bitlodine: extracting intelligence from the bitcoin network," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 8437, 2014, pp. 457–468.
- [6] K. Liao, Z. Zhao, A. Doupe, and G.-J. Ahn, "Behind closed doors: measurement and analysis of CryptoLocker ransoms in Bitcoin," in *2016 APWG Symposium on Electronic Crime Research (eCrime)*, Jun. 2016, vol. 2016-June, pp. 1–13, doi: 10.1109/ECRIME.2016.7487938.
- [7] S. Bistarelli, M. Parrocini, and F. Santini, "Visualising bitcoin flows of ransomware: WannaCry one week later," *CEUR Workshop Proceedings*, vol. 2058, 2018.
- [8] C. Brenig, R. Accorsi, and G. Möller, "Economic analysis of cryptocurrency backed money laundering," *23rd European Conference on Information Systems, ECIS 2015*, vol. 2015-May, 2015.
- [9] M. Moser, R. Bohme, and D. Breuker, "An inquiry into money laundering tools in the Bitcoin ecosystem," in *2013 APWG eCrime Researchers Summit*, Sep. 2013, pp. 1–14, doi: 10.1109/eCRS.2013.6805780.
- [10] T. Moore, J. Han, and R. Clayton, "The postmodern ponzi scheme: empirical analysis of high-yield investment programs," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 7397 LNCS, pp. 41–56, 2012, doi: 10.1007/978-3-642-32946-3_4.
- [11] M. Vasek and T. Moore, "There's no free lunch, even using bitcoin: tracking the popularity and profits of virtual currency scams," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 8975, 2015, pp. 44–61.
- [12] T. Moore, "The promise and perils of digital currencies," *International Journal of Critical Infrastructure Protection*, vol. 6, no. 3–4, pp. 147–149, Dec. 2013, doi: 10.1016/j.ijcip.2013.08.002.
- [13] M. Möser, R. Böhme, and D. Breuker, "Towards risk scoring of bitcoin transactions," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 8438, 2014, pp. 16–32.
- [14] L. Vinet and A. Zhedanov, "A 'missing' family of classical orthogonal polynomials," *Brooklyn Journal of International Law*, vol. 39, p. 829, Nov. 2010, doi: 10.1088/1751-8113/44/8/085201.
- [15] H. Darmawan, M. Yuliana, and M. Z. Samson Hadi, "GRU and XGBoost performance with hyperparameter tuning using GridSearchCV and bayesian optimization on an IoT-based weather prediction system," *International Journal on Advanced Science, Engineering and Information Technology*, vol. 13, no. 3, pp. 851–862, Jun. 2023, doi: 10.18517/ijaseit.13.3.18377.
- [16] W. Chen et al., "SADPonzi: detecting and characterizing ponzi schemes in ethereum smart contracts," *Proceedings of the ACM on Measurement and Analysis of Computing Systems*, vol. 5, no. 2, pp. 1–30, Jun. 2021, doi: 10.1145/3460093.
- [17] M. Wang and J. Huang, "Detecting ethereum ponzi schemes through opcode context analysis and oversampling-based AdaBoost algorithm," *Computer Systems Science and Engineering*, vol. 47, no. 1, pp. 1023–1042, 2023, doi: 10.32604/csse.2023.039569.
- [18] A. Aljofey, Q. Jiang, and Q. Qu, "A supervised learning model for detecting ponzi contracts in ethereum blockchain," in *Communications in Computer and Information Science*, vol. 1563 CCIS, 2022, pp. 657–672.
- [19] A. Aljofey, A. Rasool, Q. Jiang, and Q. Qu, "A feature-based robust method for abnormal contracts detection in ethereum blockchain," *Electronics*, vol. 11, no. 18, p. 2937, Sep. 2022, doi: 10.3390/electronics11182937.
- [20] C. Xu, R. Xiong, X. Shen, L. Zhu, and X. Zhang, "How to find a bitcoin mixer: a dual ensemble model for bitcoin mixing service detection," *IEEE Internet of Things Journal*, vol. 10, no. 19, pp. 17220–17230, 2023, doi: 10.1109/IIOT.2023.3275158.
- [21] G. Ibba, G. A. Pierro, and M. Di Francesco, "Evaluating machine-learning techniques for detecting smart ponzi schemes," in *2021 IEEE/ACM 4th International Workshop on Emerging Trends in Software Engineering for Blockchain (WETSEB)*, May 2021, pp. 34–40, doi: 10.1109/WETSEB52558.2021.00012.
- [22] S. Yu, J. Jin, Y. Xie, J. Shen, and Q. Xuan, "Ponzi scheme detection in ethereum transaction network," in *Communications in Computer and Information Science*, vol. 1490 CCIS, 2021, pp. 175–186.
- [23] Y. Zhang, W. Yu, Z. Li, S. Raza, and H. Cao, "Detecting ethereum ponzi schemes based on improved LightGBM algorithm," *IEEE Transactions on Computational Social Systems*, vol. 9, no. 2, pp. 624–637, Apr. 2022, doi: 10.1109/TCSS.2021.3088145.
- [24] S. Fan, S. Fu, H. Xu, and C. Zhu, "Expose your mask: smart ponzi schemes detection on blockchain," in *2020 International Joint Conference on Neural Networks (IJCNN)*, Jul. 2020, pp. 1–7, doi: 10.1109/IJCNN48605.2020.9207143.
- [25] Y. Lou, Y. Zhang, and S. Chen, "Ponzi contracts detection based on improved convolutional neural network," in *2020 IEEE International Conference on Services Computing (SCC)*, Nov. 2020, pp. 353–360, doi: 10.1109/SCC49832.2020.00053.





- [26] Z. Zheng, W. Chen, Z. Zhong, Z. Chen, and Y. Lu, "Securing the Ethereum from smart ponzi schemes: identification using static features," *ACM Transactions on Software Engineering and Methodology*, vol. 32, no. 5, pp. 1–28, Sep. 2023, doi: 10.1145/3571847.
- [27] S. Zhang, T. Lan, L. Wang, S. Xu, and W. Shao, "Ethereum ponzi scheme detection based on PD-SECR," *Security and Communication Networks*, vol. 2022, pp. 1–15, Sep. 2022, doi: 10.1155/2022/2316310.
- [28] R. Liang *et al.*, "PonziGuard: detecting ponzi schemes on ethereum with contract runtime behavior graph (CRBG)," in *Proceedings of the IEEE/ACM 46th International Conference on Software Engineering*, Feb. 2024, pp. 1–12, doi: 10.1145/3597503.3623318.
- [29] I. J. Onu, A. E. Omolara, M. Alawida, O. I. Abiodun, and A. Alabdultif, "Detection of Ponzi scheme on Ethereum using machine learning algorithms," *Scientific Reports*, vol. 13, no. 1, p. 18403, Oct. 2023, doi: 10.1038/s41598-023-45275-0.
- [30] Polarwolf, "Ponzi scheme contracts on ethereum," 2020, [Online]. Available: [https://www.kaggle.com/datasets/polarwolf/ponzi-scheme-contracts-on-ethereum?select=Ponzi contracts.csv](https://www.kaggle.com/datasets/polarwolf/ponzi-scheme-contracts-on-ethereum?select=Ponzi%20contracts.csv).
- [31] B. Bin Jia and M. L. Zhang, "Multi-dimensional classification via sparse label encoding," *Proceedings of Machine Learning Research*, vol. 139, pp. 4917–4926, 2021.
- [32] A. Q. Abdulghani, O. N. Ucan, and K. M. A. Alheeti, "Credit card fraud detection using XGBoost algorithm," in *2021 14th International Conference on Developments in eSystems Engineering (DeSE)*, Dec. 2021, vol. 2021-Decem, pp. 487–492, doi: 10.1109/DeSE54285.2021.9719580.
- [33] B. Cui and G. Wang, "Ponzi scheme detection based on CNN and BiGRU combined with attention mechanism," in *2024 27th International Conference on Computer Supported Cooperative Work in Design (CSCWD)*, May 2024, pp. 1852–1857, doi: 10.1109/CSCWD61410.2024.10580692.
- [34] Z. Ferdoush, B. N. Mahmud, A. Chakrabarty, and J. Uddin, "A short-term hybrid forecasting model for time series electrical-load data using random forest and bidirectional long short-term memory," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 11, no. 1, p. 763, Feb. 2021, doi: 10.11591/ijece.v11i1.pp763-771.

BIOGRAPHIES OF AUTHORS







Fahad Hossain     is a graduate student at the Department of Computer and Information Science in the Florida International University (FIU), Miami, Florida, United States of America. Prior to starting M.S.C at FIU, Hossain completed his Bachelor's degree in Computer Science and Engineering from Brac University, Dhaka, Bangladesh. His research interests include financial fraud detection, blockchain, and cloud networks. He can be contacted at email: fhoss006@fiu.edu.



Mehedi Hasan Shuvo     is a B.Sc. Engineer, having graduated from the Department of Computer Science and Engineering at Dhaka University of Engineering and Technology in Gazipur, Bangladesh. He is a courteous and dedicated individual with strong technical skills, passionate about tackling challenges and creating engaging, informative content. He has three years of industrial experience in mobile application development (Android and iOS) and has worked full-time in three professors' labs at his university, which further refined his expertise. His research interests span federated learning, explainable artificial intelligence, computer vision, edge computing, machine learning, deep learning, object detection, and water and environmental sustainability. He can be contacted at email: mehedihasanshuvo.mail@gmail.com.



Jia Uddin     is an assistant professor in the AI and Big Data Department, Endicott College, Woosong University, Korea. He is an associate professor (currently on leave) in the CSE department at BRAC University, Bangladesh. He received a Ph.D. degree (Computer Engineering) from the University of Ulsan, South Korea in January 2015 and an MSc in Telecommunications from Blekinge Institute of Technology, Sweden, in 2010. He was a member of the Self-Assessment Team (SAC) of CSE, BRACU in the HEQEP project funded by the World Bank and the University of Grant Commission Bangladesh in 2016–2017. His research area includes fault diagnosis using AI, audio, and image processing. He can be contacted at email: jia.uddin@wsu.ac.kr.