International Journal of Computer Science & Communications
Vol. 11, No. 1, 2026, pp. 15-29
http://ijcsc.ielas.org/index.php/ijcsc
ISSN: 2545-4153

HIDE AND SEEK: AN ADVERSARIAL HIDING APPROACH AGAINST PHISHING DETECTION ON ETHEREUM

A Moulika[1]*, Dr. K. P. Supreethi[2]
[1]*Department of Computer Science and Engineering, JNTUH, India
Corresponding Author: anmanthulamoulika@gmail.com
[2]Department of Computer Science and Engineering, JNTUH, India
E-mail: supreethi.pujari@jntuh.ac.in

Received: 2026-08-26 Accepted: 2026-09-12 Published online: 2026-09-27

Abstract

The increasing use of Ethereum for cryptocurrency transactions and decentralized applications has created new security challenges, particularly phishing attacks involving fraudulent accounts and malicious transactions. Since blockchain transactions are generally irreversible, identifying suspicious account behavior at an early stage is important for reducing potential financial losses. This paper presents a machine learning-based framework for detecting phishing accounts using Ethereum transaction behavioral data. The framework performs data preprocessing, feature extraction, supervised classification, and performance evaluation using Logistic Regression, Support Vector Machine, Random Forest, and Gradient Boosting Decision Tree, using transaction-level attributes such as sender and receiver addresses, transferred value, and contract information. The models are evaluated using Accuracy, Precision, Recall, F1-Score, and confusion matrix analysis. Among the four classifiers, Random Forest achieves the strongest performance, with 97.19% accuracy and a 97.79% F1-score, while Logistic Regression, SVM, and Gradient Boosting Decision Tree provide comparatively lower but still meaningful detection performance. Feature-importance analysis is also used to identify the transaction characteristics that contribute most to phishing classification. In addition, adversarial hiding is considered to examine the robustness of the trained models when malicious behavior is deliberately modified to resemble legitimate activity; the resulting Attack Success Rate ranges from 68.14% for Logistic Regression to 100% for Random Forest and Gradient Boosting Decision Tree, showing that high accuracy under normal conditions does not guarantee resistance to adaptive attackers. The proposed framework provides a systematic approach for comparing machine learning models for Ethereum phishing detection and analyzing their robustness against evolving attack behavior, offering a foundation for developing more adaptive and reliable blockchain security systems.

Keywords: Ethereum, blockchain security, phishing detection, machine learning, transaction analysis, Random Forest, Gradient Boosting, adversarial attacks.

I. INTRODUCTION

Blockchain technology provides a decentralized mechanism for recording and verifying digital transactions without depending on a single central authority. Ethereum has become an important blockchain platform because it supports cryptocurrency

transactions, smart contracts, and decentralized applications. The availability of Ethereum transaction records also provides an opportunity to study account behavior and identify suspicious activities using data-driven techniques (Zheng et al., 2021).

Phishing is one of the important security threats in cryptocurrency ecosystems. Attackers may use fraudulent accounts, malicious transactions, fake services, or deceptive activities to obtain digital assets from users. Unlike conventional financial transactions, confirmed blockchain transfers are difficult to reverse, which can make successful phishing attacks particularly harmful (Chen et al., 2020; Yuan et al., 2020).

The large volume of Ethereum transactions makes manual identification of suspicious accounts difficult. Traditional detection mechanisms based on predefined rules or previously identified addresses may also have limitations when attackers change their behavior. Therefore, machine learning techniques can be used to learn behavioral patterns from transaction data and classify accounts according to their likelihood of being associated with phishing activity (Chen et al., 2020; Yuan et al., 2020; Wu et al., 2022; Xia et al., 2022; Huang et al., 2022; Liu et al., 2022; Farrugia et al., 2020).

Previous studies have investigated Ethereum phishing detection using transaction-based features, network representations, and graph learning techniques (Wu et al., 2022; Xia et al., 2022; Huang et al., 2022; Liu et al., 2022; Farrugia et al., 2020). These approaches demonstrate that account interactions and transaction characteristics contain useful information for identifying malicious behavior. However, attackers may deliberately alter their transaction patterns to appear more similar to legitimate accounts, creating an additional challenge for detection systems (Wen et al., 2021).

This study develops a comparative machine learning framework using Logistic Regression (LR), Support Vector Machine (SVM), Random Forest (RF), and Gradient Boosting Decision Tree (GBDT). The framework uses transaction behavioral characteristics to classify Ethereum accounts and evaluates the models using standard classification metrics. In addition, feature importance and adversarial hiding analysis are incorporated to provide further insight into model behavior and robustness.

The main contributions of this study are:

1) A machine learning framework is developed for Ethereum phishing-account detection using transaction behavioral information.

2) Four supervised learning algorithms are comparatively evaluated under a common experimental setup.

3) Standard classification metrics are used to measure detection performance.

4) Feature-importance analysis is used to identify important transaction characteristics.

5) Adversarial hiding analysis is performed to examine the robustness of the trained classifiers.

The remainder of this paper is organized as follows. Section II presents related research on Ethereum transaction analysis, phishing detection, graph-based methods, and adversarial attacks. Section III describes the proposed methodology. Section IV discusses the obtained results. Section V concludes the study and presents future research directions.

II. RELATED WORKS

A. Ethereum Transaction Analysis

Ethereum provides publicly accessible transaction records that can be analyzed to understand interactions between blockchain accounts. Previous research has represented transaction records as networks, where accounts are modeled as nodes and transactions form connections between them (Lin et al., 2020a; Lin et al., 2020b; Wu et al., 2021). Such representations allow account relationships, transaction frequency, transaction values, and changes in transaction behavior to be investigated.

Temporal and weighted transaction information can provide additional behavioral information for blockchain analysis. Transaction-network representations have therefore been applied to study the evolution of Ethereum accounts and their interactions (Lin et al., 2020a; Lin et al., 2020b; Breiman, 2001). These studies demonstrate that transaction records can provide useful features for identifying abnormal and potentially malicious activity.

B. Ethereum Phishing Detection

Machine learning has increasingly been applied to detect phishing accounts on Ethereum. Transaction-based approaches analyze account activity and transaction characteristics to distinguish suspicious accounts from legitimate ones (Chen et al., 2020; Yuan et al., 2020). Network-embedding approaches additionally use relationships between connected accounts to obtain richer representations of blockchain behavior (Wu et al., 2022).

Attributed graph representations have also been explored by combining account attributes with neighborhood information (Xia et al., 2022). These approaches demonstrate that information from an account's surrounding transaction environment can improve phishing detection. However, more complex graph-based methods may

require additional computational resources when applied to large blockchain transaction networks.

Other research has investigated the detection of illicit Ethereum accounts and related fraudulent activities using machine learning and network-based representations (Farrugia et al., 2020; Chen et al., 2018). These studies indicate that automated analysis can improve the identification of suspicious blockchain behavior compared with purely manual investigation.

C. Graph-Based and Adversarial Approaches

Graph neural networks and graph convolutional approaches have been investigated for Ethereum account classification because blockchain transactions naturally form interconnected structures (Huang et al., 2022; Liu et al., 2022). By learning information from neighboring accounts, graph-based models can capture relationships that may not be represented by independent transaction features.

However, blockchain detection systems must also consider adversarial behavior. Attackers can modify transaction patterns or network relationships to make malicious accounts appear legitimate. Research on adversarial attacks against graph-based learning has shown that carefully designed modifications can affect model predictions (Sun et al., 2018; Dai et al., 2018; Zügner et al., 2019). Ethereum-specific research has similarly highlighted transaction-based hiding strategies against phishing detection systems (Wen et al., 2021).

These observations indicate that classification accuracy under normal conditions alone is not sufficient to characterize a blockchain security model. A useful detection framework should also examine how its performance changes when malicious users attempt to evade detection.

D. Research Gap

Existing research demonstrates the effectiveness of transaction-based, network-based, and graph-based techniques for Ethereum phishing detection. However, several challenges remain. Complex graph models may require substantial computational resources, while conventional classifiers may be affected by changing phishing behavior. In addition, adversarial hiding can reduce detection performance when malicious accounts modify their observable transaction characteristics (Wen et al., 2021; Sun et al., 2018; Dai et al., 2018; Zügner et al., 2019).

Therefore, this study focuses on a comparative machine learning framework based on Ethereum transaction behavioral features. Logistic Regression, SVM, Random Forest, and GBDT are evaluated under a common experimental procedure, followed by feature-importance analysis and adversarial hiding evaluation. This combination provides both a

performance comparison and an assessment of robustness against modified phishing behavior.

III. PROPOSED METHODOLOGY

A. Overall Proposed Framework

The proposed framework is designed to classify Ethereum accounts as legitimate or phishing based on transaction behavioral information. The complete process consists of dataset preparation, preprocessing, feature extraction, model training, classification, performance evaluation, and adversarial analysis.

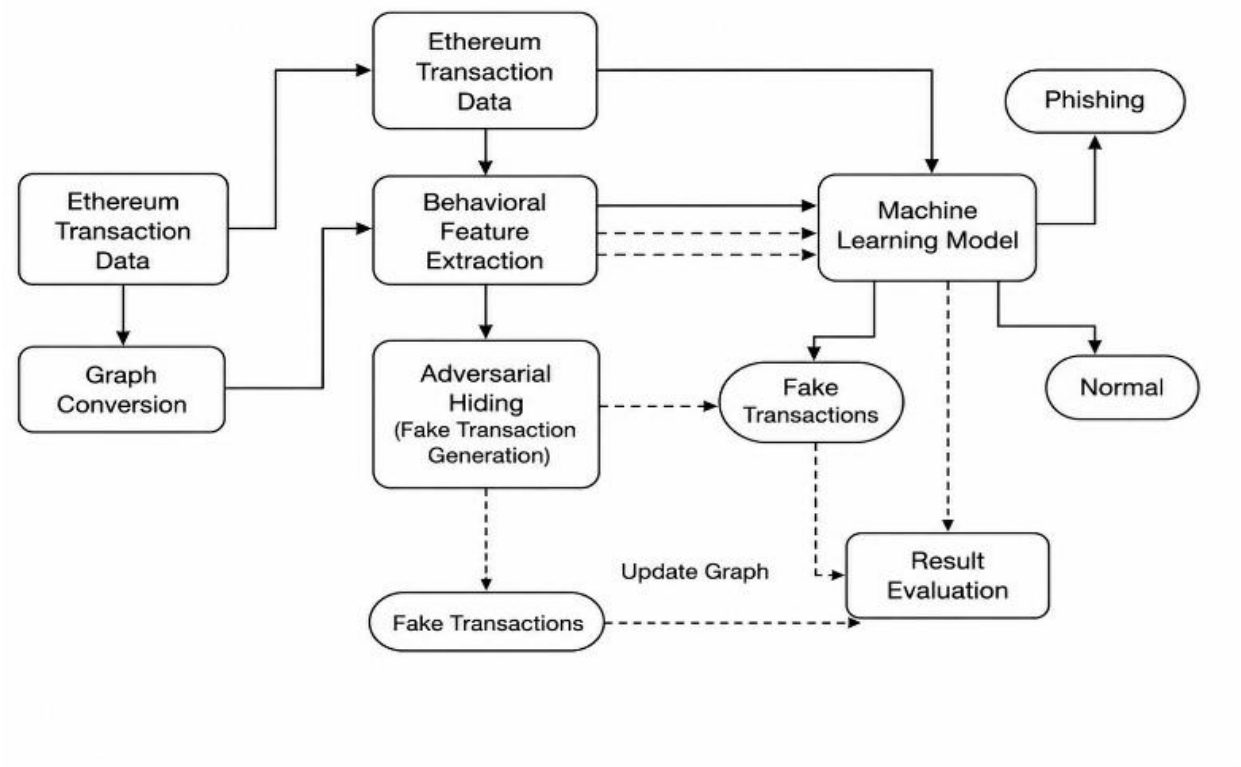

Fig. 1. *Proposed framework for Ethereum phishing detection.*

The input Ethereum transaction data is first cleaned and transformed into a suitable machine learning representation. Relevant behavioral characteristics are then extracted and supplied to multiple supervised learning models. The predictions generated by the models are evaluated using standard classification metrics. Finally, adversarial hiding is applied to examine whether modified phishing behavior affects the detection capability of the classifiers.

B. Dataset

The study uses Ethereum transaction data containing transaction-level information associated with blockchain accounts. The dataset provides behavioral information that can be processed to distinguish legitimate activity from phishing-related activity.

The available transaction information includes fields associated with transaction identification, block information, timestamp, sender and receiver addresses, transferred value, contract information, input information, and the corresponding class label.

Table 1

Main Transaction Attributes Used in the Study

Attribute	Description
Tx Hash	Transaction identifier
Block Height	Block containing the transaction
Time Stamp	Transaction time
From	Sender account
To	Receiver account
Value	Transferred amount
Contract Address	Associated contract information
Input	Transaction input information
Class	Account/activity class

The exact attributes used for model training are determined after preprocessing and feature selection.

C. Data Preprocessing

The raw transaction data is first examined to identify irrelevant fields, missing values, and inconsistent entries. Non-informative attributes are removed where necessary, while missing values are handled using the preprocessing procedure implemented in the project. The cleaned data is then converted into a consistent structure for feature extraction.

This step reduces unnecessary information and prepares the transaction records for machine learning. Maintaining the same preprocessing procedure for all classifiers also provides a fair basis for model comparison.

D. Feature Extraction

After preprocessing, transaction-level information is transformed into behavioral features representing the activity of Ethereum accounts. These characteristics include information related to transaction frequency, transaction volume, transfer behavior, and account activity.

The extracted features provide the input representation for the classification models. By using behavioral characteristics rather than relying only on fixed phishing addresses, the framework attempts to identify suspicious patterns that may also occur in previously unseen accounts.

E. Machine Learning Models

Four supervised learning algorithms are used in the proposed framework.

1) Logistic Regression: Logistic Regression provides a baseline classification approach for distinguishing the two account classes (Kleinbaum et al., 2002).

2) Support Vector Machine: SVM identifies a decision boundary that separates the classes and is suitable for classification problems involving multiple behavioral features.

3) Random Forest: Random Forest combines multiple decision trees and uses their collective predictions for classification. Its ensemble structure allows it to model nonlinear relationships and reduce the influence of individual noisy observations (Breiman, 2001).

4) Gradient Boosting Decision Tree: GBDT constructs decision trees sequentially, with each subsequent tree focusing on errors made by earlier trees. This allows complex relationships within the transaction features to be captured (Friedman, 2001).

F. Model Training and Classification

The processed dataset is divided into training and testing subsets using a stratified split. The training portion is used to develop the four classifiers, while the testing portion is reserved for evaluating their predictions.

Each model receives the same prepared feature representation. The resulting predictions classify the evaluated accounts into legitimate or phishing categories. The common experimental setup ensures that the comparison reflects differences between the learning algorithms rather than differences in data preparation.

G. Performance Evaluation

The models are evaluated using Accuracy, Precision, Recall, and F1-Score.

$$Accuracy = (TP + TN) / (TP + TN + FP + FN)$$

$$Precision = TP / (TP + FP)$$

$$Recall = TP / (TP + FN)$$

$$F1\text{-}Score = 2 \times (Precision \times Recall) / (Precision + Recall)$$

where TP, TN, FP, and FN represent true positives, true negatives, false positives, and false negatives, respectively.

International Journal of Computer Science & Communications
Vol. 11, No. 1, 2026, pp. 15-29
http://ijcsc.ielas.org/index.php/ijcsc
ISSN: 2545-4153

H. Adversarial Hiding Analysis

To evaluate robustness, the framework considers an adversarial scenario in which phishing behavior is modified to become more similar to legitimate transaction behavior. The modified samples are provided to the trained models, and the resulting performance is compared with the original test performance.

The performance difference is calculated as:

Performance Drop = Performance (Normal) − Performance (Adversarial)

A smaller performance reduction indicates better resistance to the simulated hiding behavior.

IV. RESULTS AND DISCUSSION

The proposed framework was evaluated using the selected Ethereum transaction dataset and four machine learning models: Logistic Regression (LR), Support Vector Machine (SVM), Random Forest (RF), and Gradient Boosting Decision Tree (GBDT). The models were compared using Accuracy, Precision, Recall, and F1-Score. In addition to normal classification, adversarial hiding analysis was performed to examine whether phishing accounts could avoid detection after their transaction behavior was modified.

A. Classification Performance

The trained models were tested using unseen Ethereum transaction records. Accuracy measures the overall percentage of correctly classified accounts, while Precision indicates how many accounts predicted as phishing were actually phishing. Recall measures the ability to identify phishing accounts, and F1-Score provides a balanced measure of Precision and Recall.

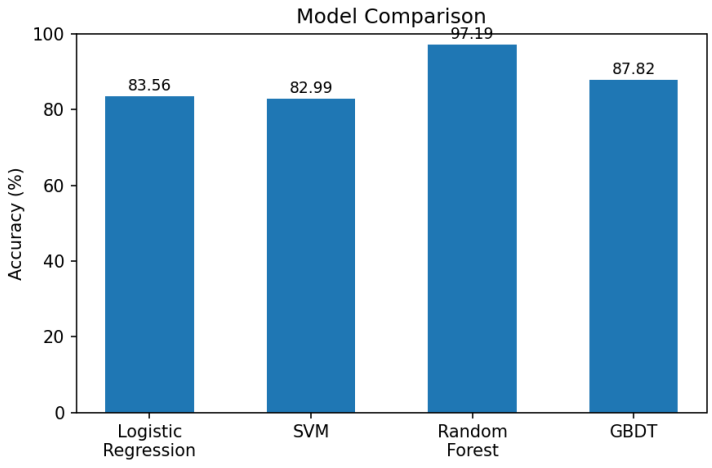

Fig. 2. *Performance comparison of the machine learning models*.

International Journal of Computer Science & Communications
Vol. 11, No. 1, 2026, pp. 15-29
http://ijcsc.ielas.org/index.php/ijcsc
ISSN: 2545-4153

The experimental results show that the machine learning models can effectively distinguish phishing and legitimate accounts. Among the evaluated classifiers, the ensemble-based models, particularly Random Forest and GBDT, provide strong performance because they can capture nonlinear relationships within transaction behavioral features. The comparison also helps identify the most suitable model for the proposed phishing detection framework.

Table 2

Performance Comparison of Machine Learning Models

Metrics	Logistic Regression	SVM	Random Forest	GBDT
Accuracy (%)	83.56	82.99	97.19	87.82
Precision (%)	76.56	66.81	99.80	75.59
Recall (%)	71.22	66.81	95.85	75.59
F1-Score (%)	73.79	66.81	97.79	75.59

B. Confusion Matrix Analysis

A confusion matrix was used to analyze the classification results in more detail. It shows the number of correctly and incorrectly classified legitimate and phishing accounts. True positives represent correctly detected phishing accounts, while false negatives represent phishing accounts that were incorrectly classified as legitimate.

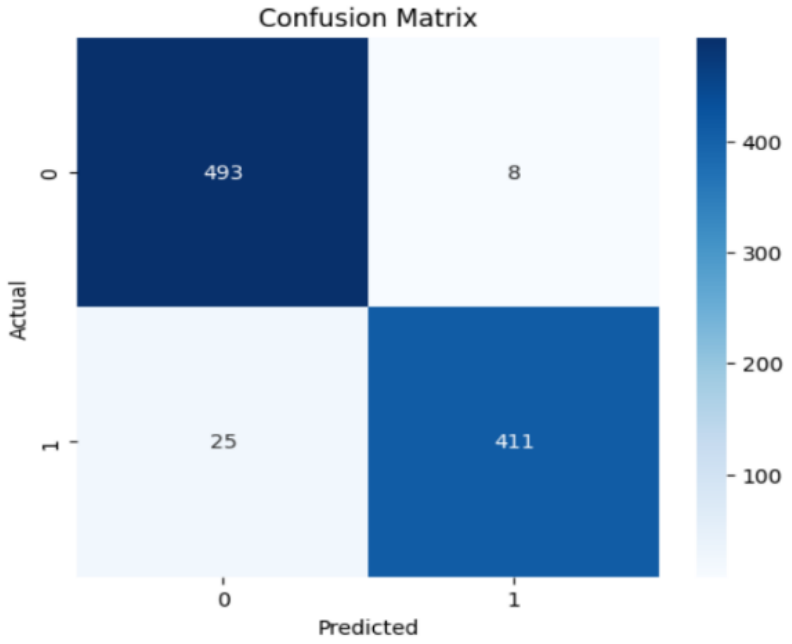

Fig. 3. *Confusion matrix of the best-performing model.*

International Journal of Computer Science & Communications
Vol. 11, No. 1, 2026, pp. 15-29
http://ijcsc.ielas.org/index.php/ijcsc
ISSN: 2545-4153

A higher number of correctly classified samples and a lower number of false negatives indicate better phishing detection capability. The confusion matrix therefore provides additional information that cannot be obtained from Accuracy alone.

C. Feature Importance Analysis

Feature importance analysis was performed to determine which transaction characteristics contributed most to phishing classification. Features with higher importance have a greater influence on the model's prediction.

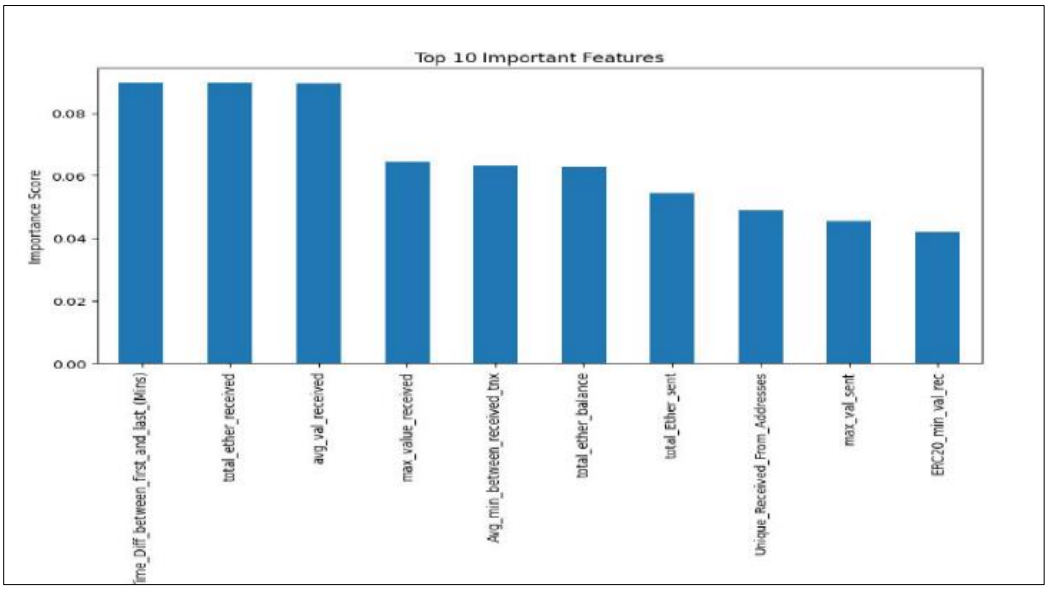

Fig. 4. *Feature importance of Ethereum transaction characteristics.*

The analysis helps understand the behavioral patterns associated with phishing accounts and provides additional insight into how the machine learning models make their predictions.

D. Adversarial Hiding Analysis

In addition to normal classification, an adversarial experiment was performed to examine the robustness of the proposed detection framework. In this experiment, some phishing accounts are modified to behave more like legitimate accounts by changing their transaction-related features.

The modified phishing samples are then given to the trained models again. The objective is to determine whether the models can still correctly identify these hidden phishing accounts.

The adversarial evaluation consists of the following steps:

1) Select phishing accounts from the test data.

2) Modify selected transaction features to resemble legitimate behavior.

3) Apply the modified samples to the trained classifiers.

4) Record the number of phishing accounts detected and missed.

5) Compare normal and adversarial detection performance.

This experiment represents a realistic situation in which attackers attempt to hide their malicious behavior from machine learning-based detection systems.

E. Attack Success Rate (ASR)

Attack Success Rate (ASR) is used to measure how effectively the adversarial hiding technique causes phishing accounts to escape detection.

The ASR is calculated as:

$$ASR = (N\_escaped / N\_targeted) \times 100$$

where N_escaped is the number of phishing accounts incorrectly classified as legitimate after the attack, and N_targeted is the total number of phishing accounts subjected to the hiding attack.

A higher ASR indicates that the hiding attack was more successful, meaning more phishing accounts were able to escape detection. Conversely, a lower ASR indicates that the machine learning model is more resistant to adversarial hiding.

The Normal Accuracy values reported below are the same baseline results presented in Table II; they are repeated here alongside the adversarial results so that the Performance Drop and ASR for each model can be read directly from a single table.

Table 2

Adversarial Hiding Performance

Model	Normal Accuracy (%)	Adversarial Accuracy (%)	Performance Drop (%)	ASR (%)
Logistic Regression	83.56	79.83	3.73	68.14
SVM	82.99	76.78	6.21	97.95
Random Forest	97.19	87.47	9.72	100.00
GBDT	87.82	80.45	7.37	100.00

International Journal of Computer Science & Communications
Vol. 11, No. 1, 2026, pp. 15-29
http://ijcsc.ielas.org/index.php/ijcsc
ISSN: 2545-4153

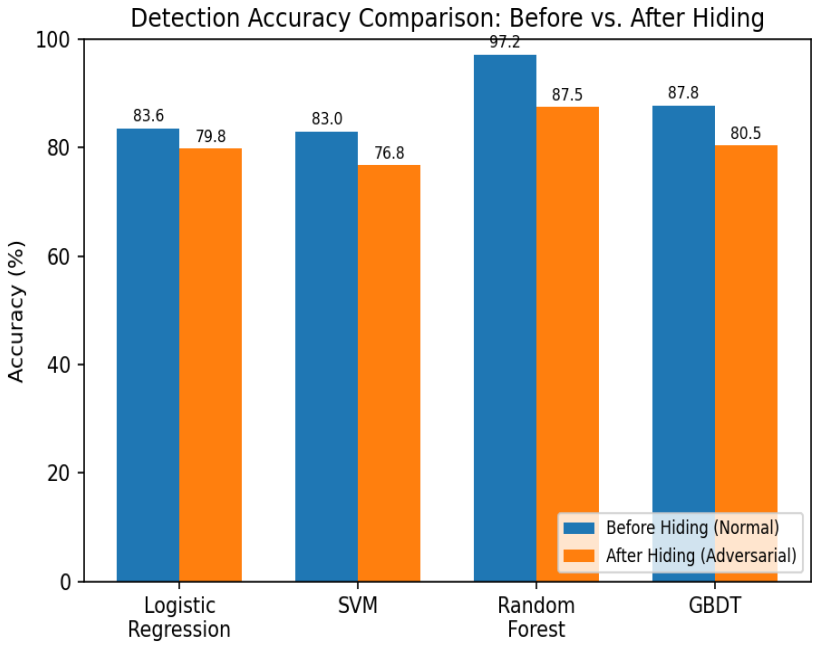

Fig. 5. *Detection accuracy of each model before and after the adversarial hiding attack, corresponding to the Normal and Adversarial Accuracy columns of Table 3*

The adversarial results provide an additional measure of model reliability. A model that maintains higher detection performance and produces a lower ASR under modified phishing behavior can be considered more robust against hiding attacks. This analysis is important because high classification accuracy under normal conditions alone does not guarantee resistance to adaptive attackers.

F. Overall Discussion

The experimental evaluation demonstrates that machine learning can effectively support Ethereum phishing detection using transaction behavioral information. The comparison of multiple classifiers provides a clear understanding of their relative strengths, while feature-importance analysis helps identify influential transaction characteristics.

The adversarial experiment further shows why robustness should be considered when developing blockchain security systems. Phishing attackers may deliberately modify their transaction behavior to resemble legitimate accounts, and such changes can reduce detection performance. Therefore, evaluating both normal classification and adversarial conditions provides a more comprehensive assessment of the proposed framework.

Overall, the combination of classification performance, feature analysis, confusion-matrix evaluation, and Attack Success Rate (ASR) gives a stronger evaluation of the Ethereum phishing detection system.

V. CONCLUSION AND FUTURE WORK

The increasing use of Ethereum for digital transactions and decentralized applications has created a growing need for effective phishing detection mechanisms. This study developed a machine learning-based framework for identifying potentially malicious Ethereum accounts using transaction-related behavioral information. The proposed approach follows a structured process involving data preprocessing, feature extraction, model training, classification, and performance evaluation.

Four supervised machine learning algorithms—Logistic Regression, Support Vector Machine (SVM), Random Forest, and Gradient Boosting Decision Tree (GBDT)—were implemented and compared. Their performance was evaluated using Accuracy, Precision, Recall, F1-Score, and Confusion Matrix. The experimental analysis indicates that ensemble-based models, particularly Random Forest and GBDT, are effective in learning complex transaction patterns and provide strong phishing detection performance compared with conventional classifiers.

The framework also includes adversarial hiding analysis to examine the effect of modified phishing behavior on detection performance. In this scenario, selected phishing accounts are made to appear more similar to legitimate accounts by modifying their transaction-related characteristics. The Attack Success Rate (ASR) is used to measure how many targeted phishing accounts successfully evade detection. This analysis provides an additional measure of the robustness of the proposed models beyond normal classification accuracy.

Overall, the proposed framework provides a systematic approach for Ethereum phishing detection by combining behavioral analysis, machine learning classification, performance evaluation, and adversarial robustness analysis. The approach can serve as a foundation for developing more adaptive and reliable blockchain security systems capable of addressing evolving phishing threats.

Future Scope

The proposed Ethereum phishing detection framework can be further improved in terms of accuracy, scalability, real-time monitoring, and robustness. The following directions can be considered for future development:

- Graph Neural Networks (GNNs): GNNs can be used to analyze relationships between Ethereum accounts and transactions, providing better understanding of transaction networks.

• Advanced Deep Learning: Models such as LSTM, Transformer, and Autoencoder can be explored to identify complex and sequential transaction patterns.

• Real-Time Detection: The framework can be extended to monitor live Ethereum transactions and identify suspicious accounts at an early stage.

• Explainable AI (XAI): XAI techniques can provide understandable reasons for classifying an account as phishing, improving transparency and user trust.

• Cross-Blockchain Detection: The system can be extended to other blockchain platforms such as Bitcoin, Binance Smart Chain, and Solana.

• Federated Learning: Federated learning can support collaborative model training across distributed systems while reducing the need to share sensitive data.

• Adaptive Learning: Future models can be designed to continuously learn from new transaction patterns and adapt to emerging phishing strategies.

These improvements can make the proposed framework more accurate, scalable, adaptive, and resistant to evolving phishing attacks, supporting the development of a more secure blockchain ecosystem.

REFERENCES

Breiman, L. (2001). Random forests. *Machine Learning*, 45(1), 5–32.

Chen, W., Guo, X., Chen, Z., Zheng, Z., & Lu, Y. (2020). Phishing scam detection on Ethereum: Towards financial security for blockchain ecosystem. In *Proceedings of the 29th International Joint Conference on Artificial Intelligence (IJCAI)* (pp. 4506–4512). Yokohama, Japan.

Chen, W., Zheng, Z., Cui, J., Ngai, E., Zheng, P., & Zhou, Y. (2018). Detecting Ponzi schemes on Ethereum: Towards healthier blockchain technology. In *Proceedings of the World Wide Web Conference* (pp. 1409–1418). Lyon, France.

Cover, T., & Hart, P. (1967). Nearest neighbor pattern classification. *IEEE Transactions on Information Theory*, 13(1), 21–27.

Dai, H., et al. (2018). Adversarial attack on graph structured data. In *Proceedings of the International Conference on Machine Learning (ICML)* (pp. 1115–1124).

Farrugia, S., Ellul, J., & Azzopardi, G. (2020). Detection of illicit accounts over the Ethereum blockchain. *Expert Systems with Applications*, 150.

Friedman, J. H. (2001). Greedy function approximation: A gradient boosting machine. *Annals of Statistics*, 29(5), 1189–1232.

Huang, T., Lin, D., & Wu, J. (2022). Ethereum account classification based on graph convolutional network. *IEEE Transactions on Circuits and Systems II*, 69(5), 2528–2532.

Kleinbaum, D. G., Dietz, K., Gail, M., Klein, M., & Klein, M. (2002). *Logistic regression.* Springer.

Lin, D., Wu, J., Yuan, Q., & Zheng, Z. (2020a). Modeling and understanding Ethereum transaction records via a complex network approach. *IEEE Transactions on Circuits and Systems II*, 67(11), 2737–2741.

Lin, D., Wu, J., Yuan, Q., & Zheng, Z. (2020b). T-EDGE: Temporal weighted multidigraph embedding for Ethereum transaction network analysis. *Frontiers in Physics*, 8.

Liu, J., Zheng, J., Wu, J., & Zheng, Z. (2022). FA-GNN: Filter and augment graph neural networks for account classification in Ethereum. *IEEE Transactions on Network Science and Engineering*, 9(4), 2579–2588.

Sun, L., et al. (2018). Adversarial attack and defense on graph data: A survey. *arXiv preprint arXiv:1812.10528.*

Wen, H., Fang, J., Wu, J., & Zheng, Z. (2021). Transaction-based hidden strategies against general phishing detection framework on Ethereum. In *Proceedings of the IEEE International Symposium on Circuits and Systems (ISCAS)* (pp. 1–5).

Wu, J., Liu, J., Zhao, Y., & Zheng, Z. (2021). Analysis of cryptocurrency transactions from a network perspective: An overview. *Journal of Network and Computer Applications*, 190.

Wu, J., et al. (2022). Who are the phishers? Phishing scam detection on Ethereum via network embedding. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 52(2), 1156–1166.

Xia, Y., Liu, J., & Wu, J. (2022). Phishing detection on Ethereum via attributed ego-graph embedding. *IEEE Transactions on Circuits and Systems II*, 69(5), 2538–2542.

Yuan, Q., Huang, B., Zhang, J., Wu, J., Zhang, H., & Zhang, X. (2020). Detecting phishing scams on Ethereum based on transaction records. In *Proceedings of the IEEE International Symposium on Circuits and Systems (ISCAS)* (pp. 1–5).

Zheng, Z., Dai, H.-N., & Wu, J. (2021). *Blockchain intelligence: Methods, applications and challenges.* Springer.

Zügner, D., Akbarnejad, A., & Günnemann, S. (2019). Adversarial attacks on neural networks for graph data. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)* (pp. 2847–2856).