PureMetaScan: An Explainable AI Framework for Detecting Vulnerable Metaverse Smart Contracts

Ebuka Chinaechetam Nkoro [1], Jae Min Lee [2], Dong-Seong Kim [3] *
[1] ICT Convergence Research Center, *Kumoh National Institute of Technology*, Gumi, South Korea
[3] IT-Convergence Engineering, *Kumoh National Institute of Technology*, Gumi, South Korea
* NSLab Co. Ltd. *Kumoh National Institute of Technology*, Gumi, South Korea
nkorochinaechetam@gmail.com, (ljmpaul, dskim)@kumoh.ac.kr

***Abstract*—This study presents PureMetaScan, a conceptual explainable AI system emphasizing the need for trustworthy detection of vulnerable Smart Contracts (SCs) within the Metaverse. Using a deep neural network and LIME explainer for transparent explanations, our solution achieves benchmark detection performance using the *Smartbugswild* dataset. Our approach safeguards digital assets, governance, and other functionalities requiring safe SCs, fostering trust within the 507.8 billion dollar Metaverse market.**

***Index Terms*—Metaverse, Smart Contracts, Blockchain, Explainable AI, Artificial Intelligence**

I. Introduction

The Metaverse [1], which strives to become the next-generation immersive Internet with decentralized governance, is regulated through self-executing blockchain Smart Contracts (SCs) [2]. SCs play a significant role in the Metaverse, such as digital asset maintenance, and voting processes. Meanwhile, SCs, although termed *smart,* can be vulnerable to attacks such as reentrancy, arithmetic & gas issues, and access control attacks. Vulnerable SCs can pose severe risks for users and Metaverse stakeholders [3].

Recent works have employed Artificial Intelligence (AI) to enable SC vulnerability detection systems to mitigate SC security risks for improved detection [4]. While most focus on accuracy improvement, there is a major gap in trustworthy and explainable SC detection, which offers reliable, confident, and secure SC vulnerability detection for security teams [5]. Therefore, we foresee the lack of explainability solutions in Metaverse SCs, and propose ***PureMetaScan***, for trustworthy SC vulnerability detection in the Metaverse.

II. System Methodology

This section outlines the proposed *PureMetaScan* as shown in Figure 1 and algorithm 1. *PureMetaScan* integrates a preprocessing phase, a DNN model for SC classification, and a model-agnostic explainability method for trustworthy and reliable SC vulnerability in Metaverse.

The benchmark *SmartBugs Wild* dataset [6] was employed for the experiment, which has about 47,518 unique solidity files, encompassing 200,000 contracts. The step-by-step approach for the dataset preprocessing is detailed in Fig. 1, where the TF-IDF vectorization technique was employed for feature extraction of Solidity SC codes. Afterward, an autoencoder was used to reduce dimensionality before training a DNN classifier for classification. The DNN consists of a sequential neural network with three densely connected layers: an initial layer with 64 neurons, followed by a 32-neuron layer, all utilizing ReLU activation and interspersed with dropout regularization layers at rates of 0.3 and 0.2, respectively. After classification, the LIME eXplainable AI (XAI) library was employed for visualizing predicted SCs.

Algorithm 1 : **XAI Metaverse SC Detection Method**

Input: Smart contracts $\mathcal{S}$, labels $\mathcal{Y}$, AE settings θ_{AE}, DNN settings θ_{DNN}, XAI tools $\mathcal{X} = \{\text{LIME}\}$
1. Feature Extraction:
Extract features $\mathcal{F} = \{\mathbf{x}_i\}$
Split into compiler and non-compiler features
Create dataset $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}$
2. Autoencoder Compression:
Train AE: $\mathbf{z}_i = f_{\text{enc}}(\mathbf{x}_i)$
Minimize reconstruction loss:

$$\mathcal{L}_{AE} = \frac{1}{N}\sum_{i=1}^{N} \|\mathbf{x}_i - \mathcal{A}_\theta(\mathbf{x}_i)\|^2$$

3. Classification:
Train DNN on $\mathbf{z}_i$ to predict y_i
Use regularization (e.g., dropout)
4. Explanation:
for each test sample $\mathbf{z}_i$ **do**
  Generate explanation using SHAP/LIME
end for
Output: Predictions $\hat{\mathcal{Y}}$, compressed features $\mathcal{Z}$, explanations $\mathcal{E}$

III. Performance Evaluation

As shown in Table I, the SC DNN model achieved overall precision, recall, F1-score, accuracy, and MCC of **68.10%, 68.15%, 68.12%, 78.77%, and 52.19%**, respectively. These results affirm the framework's generalizability across diverse input formats. The *Clean* class shows the best accuracy (**78.80%**) and F1-score (**70.01%**). *Time Manipulation* is the best-detected vulnerability class (F1: **71.26%**, MCC: **57.56%**), while *Reentrancy* remains challenging (F1: **63.01%**, MCC: **45.34%**), likely due to its complex semantic nature. Training and inference remain efficient (**16.52s** and **0.08s**, respectively), supporting practical use.

XAI Results: Fig. 2 shows the LIME XAI prediction probability of the SC vulnerability detection model classifying a Solidity SC sample index as ***time-manipulation*** with high

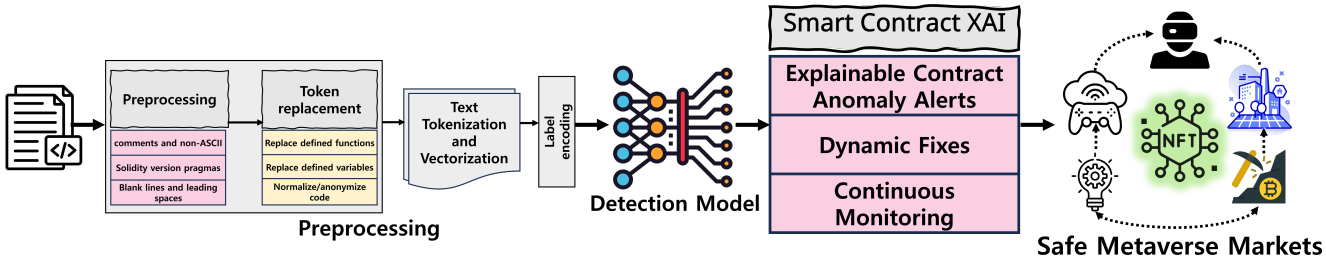

Fig. 1. illustration of the proposed *PureMetaScan*, integrating XAI for smart contract vulnerability detection in the Metaverse.

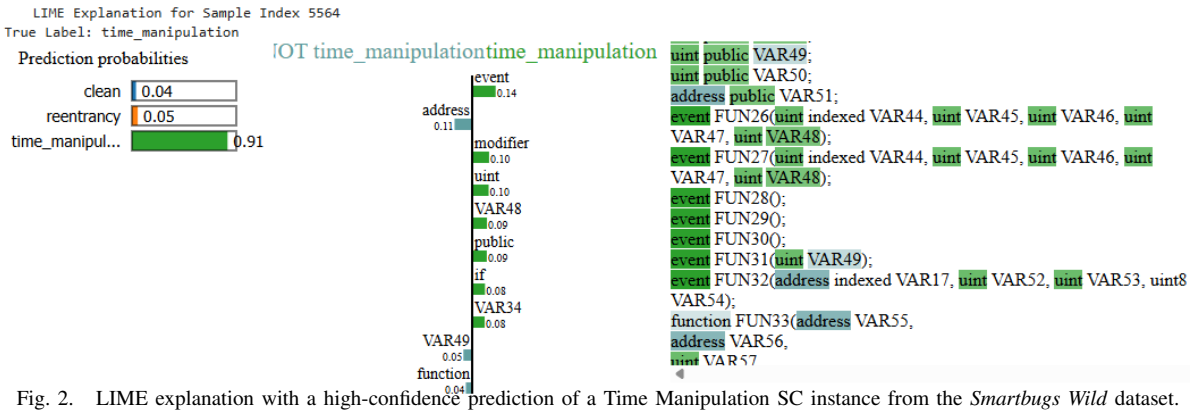

Fig. 2. LIME explanation with a high-confidence prediction of a Time Manipulation SC instance from the *Smartbugs Wild* dataset.

green bar confidence (91%). The remaining classes, such as reentrancy and clean classes, yield a very low prediction probability of 5% and 4% respectively. Top contributing feature tokens, such as event, uint, public, VAR49, and address, appear frequently and are highlighted in green, indicating their role in pushing the prediction toward *time manipulation*.

Attack Impact: Metaverse applications like live gaming require precise timing for fairness. Thus, the visual XAI detection of time manipulation prevents skewed results that favor attackers who adjust timestamps to claim rewards or leaderboards.

TABLE I
SC VULNERABILITY DETECTION PERFORMANCE USING THE SMARTBUGSWILD DATASET.

Class	Prec.	Rec.	F1	Acc.	MCC	Train / Pred (s)
Clean	69.77	70.26	70.01	78.80	53.61	
Reentrancy	63.82	62.22	63.01	76.07	45.34	16.52 / 0.08
Time Manip.	70.63	71.91	71.26	81.44	57.56	
Overall	**68.10**	**68.15**	**68.12**	**78.77**	**52.19**	

IV. CONCLUSION

The proposed *PureMetaScan* integrates XAI for SC vulnerability detection in the decentralized Metaverse. Compared to previous works in this domain, which focus mainly on accuracy improvement, the proposed scheme of this study aims towards an auditable and interpretable SCs in the Metaverse for compliance teams and developers, a gap which has remained underexplored.

ACKNOWLEDGMENT

This research was supported by the Priority Research Centers Program through the NRF funded by the MEST (2018R1A6A1A03024003) (50%) and by MSIT under the Innovative Human Resource Development for Local Intellectualization support program (IITP-2025-2020-0-01612) (50%) supervised by the IITP.

REFERENCES

[1] E. C. Nkoro, J. N. Njoku, C. I. Nwakanma, J. M. Lee, and D.-S. Kim, "Metawatch: Trends, challenges, and future of network intrusion detection in the metaverse," *IEEE Internet of Things Journal*, pp. 1–1, 2025.

[2] G. Crincoli, G. Iadarola, P. E. La Rocca, F. Martinelli, F. Mercaldo, and A. Santone, "Vulnerable Smart Contract Detection by Means of Model Checking," 2022.

[3] S. Su, Y. Tan, Y. Xue, C. Wang, H. Lu, Z. Tian, C. Shan, and X. Du, "Detecting smart contract project anomalies in metaverse," in *2023 IEEE International Conference on Metaverse Computing, Networking and Applications (MetaCom)*, 2023, pp. 524–532.

[4] P. T. Duy, N. H. Khoa, N. H. Quyen, L. C. Trinh, V. T. Kien, T. M. Hoang, and V.-H. Pham, "Vulnsense: Efficient vulnerability Detection in Ethereum Smart Contracts by Multimodal Learning with Graph Neural Network and Language Model," *International Journal of Information Security*, vol. 24, no. 1, p. 48, 2025.

[5] H. Chu, P. Zhang, H. Dong, Y. Xiao, S. Ji, and W. Li, "An Integrated Deep Learning Model for Ethereum Smart Contract Vulnerability Detection," *International Journal of Information Security*, vol. 23, p. 557–575, 2023. [Online]. Available: https://link.springer.com/article/10.1007/s10207-023-00752-5#citeas

[6] T. Durieux, J. F. Ferreira, R. Abreu, and P. Cruz, "Empirical review of automated analysis tools on 47,587 ethereum smart contracts," *Proceedings of the ACM/IEEE 42nd International conference on software engineering*, pp. 530–541, 2020.