IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 27 August 2023, accepted 4 September 2023, date of publication 8 September 2023,
date of current version 13 September 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3313448

RESEARCH ARTICLE

Anomaly VAE-Transformer: A Deep Learning Approach for Anomaly Detection in Decentralized Finance

AHYUN SONG[1], EUISEONG SEO[1], (Member, IEEE), AND HEEYOUL KIM[2]
[1]Department of Computer Science and Engineering, Sungkyunkwan University, Seoul 03063, South Korea
[2]Division of Computer Science and Engineering, Kyonggi University, Suwon 16227, South Korea

Corresponding author: Heeyoul Kim (heeyoul.kim@kyonggi.ac.kr)

This work was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education under Grant 2020R1A6A1A03040583.

ABSTRACT DeFi, a decentralized financial service based on blockchain, not only provides innovative financial services, but also poses various risks, such as the Terra Luna crash. Therefore, anomaly detection in DeFi is necessary to ensure the safety and reliability of the DeFi ecosystem. However, this is very difficult because of the complex protocol, interaction among smart contracts, and high market volatility. In this study, we propose a novel method to effectively detect anomalies in DeFi. To the best of our knowledge, this is the first study that utilizes deep learning to detect anomalies in DeFi. We propose a deep learning model, anomaly VAE-Transformer, which combines the variational autoencoder to extract local information in the short term, and the transformer, to identify dependencies between data in the long term. Based on a deep understanding of DeFi protocols, the proposed model collects and analyzes various on-chain data of Olympus DAO, a representative DeFi protocol, for extracting features suitable for anomaly detection. Then, we demonstrate the superiority of the proposed model by analyzing four anomaly cases detected successfully by the proposed model in Olympus DAO. A malicious attack attempt and structural changes in DeFi protocols can be identified quickly using the proposed method; this is expected to help protect the assets of DeFi users and improve the safety, reliability, and transparency of the DeFi market. The dataset and codes are available at *https://github.com/fialle/Anomaly-VAE-Transformer*

INDEX TERMS Anomaly detection, blockchain, deep learning, DeFi, Olympus DAO.

I. INTRODUCTION

Decentralized finance (DeFi) is a distributed finance service implemented through smart contracts on a blockchain network without using centralized financial institutions. DeFi overcomes the limitations of traditional finance, such as information asymmetry between users and institutions, high transaction fees, and delayed transactions, while heightening the transparency and accessibility of financial services and inducing various financial innovations. According to DeFi Llarma [1], the total value locked (TVL) deposited in DeFi protocols peaked at approximately $180 billion at the end of 2021, 12 times higher than that compared to $15 billion at the end of 2020. The DeFi market has been stagnant in addition to the overall decline of the cryptocurrency market that started in the first half of 2022; the TVL of DeFi protocols is around $46 billion as of May 2023.

The DeFi ecosystem is innovative and consistently growing through various financial services; however, there are also risks of abnormal transactions and fraudulent practices such as:

Flash loan attack on bZx protocol in 2020: The attacker utilized a flash loan to manipulate the market price of wBTC by exploiting a bug in bZx.

True Seigniorage Dollar (TSD) attack in 2021: The attacker abused the principles of DAO, amassing TSD tokens to gain voting power, then induced the DAO to upgrade the smart contract with malicious code.

The associate editor coordinating the review of this manuscript and approving it for publication was Rajeeb Dey.

Terra LUNA crash in 2022: This crash resulted from a rapid UST sell-off, causing oversupply of Luna and a significant price drop. The primary cause was the vulnerability in the algorithm designed to maintain the price of UST.

Smart contract attack on Yearn Finance in 2023: The attacker exploited a hardcoded misconfiguration in Yearn Finance's smart contract code.

Therefore, anomaly detection in DeFi is extremely important and necessary to protect users and improve the safety and reliability of the DeFi ecosystem.

Anomaly detection refers to the process of detecting an anomaly or outlier, which is a type of data representing patterns deviating from the normal category. In recent years, notable advancements in deep learning technology have been achieved in various fields including computer vision, natural language processing, and voice recognition, and therefore, numerous studies have focused on utilizing deep learning technology in the anomaly detection field [2], [3], [4], [5], [6], [7], [8], [9], [10], [11], [12].

However, there remain numerous challenges to realizing anomaly detection because of its distinctive characteristics. Unlike general problems, an anomaly occurs very rarely and cannot be predicted. In addition, it is defined differently based on the domain, ranging from finance and medical to smart manufacturing. Thus, a general anomaly detection model cannot be easily applied to different domains.

Recently, the blockchain technology has been under the spotlight over the past few years. Yet, the technology has not fully matured and there is a completely different technological difficulty stemming from the nature of decentralized peer-to-peer networks. Thus, research on anomaly detection using deep learning technology in a blockchain network is still in the early stages [13], [14], [15]. Past research focused on the detection of a Ponzi scheme, which is a specific type of fraud with a narrower scope than anomalies in a blockchain network [16], [17], [18], [19], [20], [21], [22], [23], [24], [25], [26], [27], [28], [29].

Thus, there is lack of research on anomaly detection in DeFi, which is a new type of a distributed financial service implemented on a blockchain network. DeFi has the technological burden of a complex structure where various protocols and tokens interact based on blockchain and smart contract technologies. Therefore, a thorough understanding of the operation principle of a blockchain and the on-chain data structure is required to analyze data related to DeFi. In addition, the market volatility of DeFi is extremely high because it is a financial ecosystem that is still in the early stages, which causes normal transaction patterns to change very rapidly or the line between abnormal and normal transactions to become unclear. Given these abovementioned reasons, detecting anomaly in DeFi is an extremely difficult and unique challenge requiring active research to obtain a solution.

In this paper, we propose an anomaly variational autoencoder (VAE)-Transformer, which is a new deep learning model for anomaly detection in DeFi. The proposed model combines a VAE for extracting local information in the short term [30] and a transformer for identifying the dependency between data in the long term [31]. The VAE encoder encodes time series daily data into low-dimensional embedding and sends it to the transformer. The transformer receives the embedding sequence and generates contextualized embeddings; the output is sent to a VAE decoder, which inputs the output of the transformer and reconstructs the daily data. The proposed model uses the difference between original and reconstructed data for anomaly detection. To the best of our knowledge, our research is the first study utilizing deep learning for anomaly detection in DeFi.

For evaluation, we applied the proposed model to Olympus decentralized autonomous organization (DAO), which is one of the largest DeFi at this moment. To this end, we collected and analyzed the transactions of Olympus DAO users related to staking, unstaking, bond creation, and bond redemption activity, the internal transactions between smart contracts invoked accordingly, and the event logs generated as execution results. Based on the analyzed on-chain data, 12 features are extracted for use in the anomaly detection model. This is unlike previous studies that detect fraud and Ponzi schemes in a blockchain using only simple transactions. Further, we calculate anomaly scores using the trained anomaly VAE-Transformer and detect anomalies in Olympus DAO. Moreover, we thoroughly analyze the four detected anomaly cases of Olympus DAO. The analysis results confirm that the proposed anomaly VAE-Transformer model can successfully detect different abnormal patterns and is suitable for anomaly detection in Olympus DAO.

The major contributions of this study are summarized below.

- To the best of our knowledge, this is the first study on anomaly detection in DeFi using deep learning technology.
- With a deep understanding of the DeFi protocol, the proposed model extracts features appropriate for anomaly detection by collecting and analyzing transactions and related various on-chain data.
- We propose the anomaly VAE-Transformer model that combines a VAE for extracting local information in the short term and a transformer for identifying dependency between data in the long term for anomaly detection in DeFi.
- The actual dataset of Olympus DAO is used. The excellence of the proposed model is proved by carefully analyzing the four cases in which the anomaly VAE-Transformer model successfully detects anomalies in Olympus DAO.

The remainder of this manuscript is organized as follows: In Section II, DeFi and Olympus DAO are examined in detail and previous studies on anomaly detection using deep learning are reviewed. In Section III, data collection and feature extraction for anomaly detection in DeFi are analyzed, and the proposed anomaly VAE-Transformer model

is explained. Section IV explains the implementation of the proposed model and the experiment on anomaly detection in Olympus DAO, and the results of anomaly detection are analyzed in detail. Finally, Section V concludes the findings of this research.

II. RELATED WORK

A. DEFI AND OLYMPUS DAO

DeFi, or decentralized finance, is a distributed finance service implemented through smart contracts without using centralized financial institutions on a blockchain network. Blockchain is a type of distributed ledger technology where anyone can read data but not manipulate it because the block data are stored in a decentralized peer-to-peer storage system. All nodes participating in a blockchain share the same records, requiring agreement among all nodes. Therefore, although a higher number of nodes in a blockchain results in inefficiency, the essential nature of decentralization is heightened because data are shared by more nodes. Thus, a blockchain has the potential to offer diverse services that differ from conventional centralized services because of the decentralization of data, server, and decision making.

In traditional finance, centralized financial companies manage and control the overall processes of financial services. That is, financial companies plan financial services, determine the terms of the offered products, and explain the details of the services to customers. Then, financial services are provided to customers by implementing the matters specified in terms and conditions. All processes are executed by the independent systems of each financial company using which all information including transaction history is recorded and managed. Consequently, customers can only obtain very limited information compared to that accessible to financial companies, and they end up paying high financial charges to these companies. Further, the safety of financial companies, where all information related to financial services is stored, is directly related to the safety of financial services.

In contrast, DeFi enables direct financial transactions among users without the need for a centralized agency by using blockchain and smart contract technology. A blockchain network plays the role of a platform in DeFi; on this network, anyone can create financial services with their own rules, and other uses who agree with such rules can freely use the relevant service without the permission of the person who created the services. All information related to DeFi including financial transactions is transparently recorded in the nodes of the blockchain network, and the DeFi service is executed through smart contracts for which the rules of the service are programmed. DeFi has been gaining considerable research interest as an alternative to traditional finance entailing structural issues such as information asymmetry between users and financial companies, high transaction fees, and delayed transactions.

The DeFi ecosystem provides traditional financial services such as deposits, loans, trade, insurance, asset management, and derivatives. Further, the DeFi ecosystem provides creative, innovative, and converged services by freely connecting and utilizing other DeFi services. Some notable DeFi services include MakerDAO, Compound, and Aave in the credit/lending field, Uniswap in the decentralized exchange (DEX) field, Yield and Synthetix in the derivatives field, and Nexus Mutual in the insurance field.

Among the various DeFi services, Olympus DAO, which is a representative service of DeFi 2.0, is examined in detail in this study. Olympus DAO is an Ethereum-based decentralized reserve currency protocol launched in March 2021, and it first proposed protocol owned liquidity (POL), which differentiates it from conventional DeFi services. Further, Olympus DAO has recently received increasing attention as an approach that can provide a higher yield than that of traditional financial services, guarantee a minimum value of OHM token as treasury owned assets, and participate in decision making through a DAO.

The minimum value of an OHM token issued by Olympus DAO is guaranteed by the assets deposited in the treasury. In other words, the treasury of Olympus DAO is used to provide liquidity and guarantee the values of the OHM tokens. The Olympus DAO treasury is operated by a DAO, and it is safely managed against hacking attempts by applying the multisignature wallet technology. The assets in the Olympus DAO treasury can increase in value through certain activities such as receiving rewards by providing liquidity when OHM token transactions increase within DEX or by transferring the difference between the sales and issuance prices of bonds through bonding sales.

Staking and bonding are two core mechanisms of Olympus DAO. The Olympus DAO participants can earn interest through staking and purchase OHM tokens at a discounted price through bonding. Further, the Olympus DAO protocol can secure safety by directly owning liquidity and treasury assets through staking and bonding.

Bonding plays an important role in increasing the assets owned by the treasury. Bonders (bonding participants) can purchase an OHM token at a discounted price from the market price. Further, bonders can participate in staking by owing OHM tokens or earn profits by selling the token after a certain period (five days by default). The protocol receives assets from bonders and transfers them to the treasury assets. Stakers stake OHM tokens in a protocol and receive the rebase rewards. Olympus DAO provides a high yield to stakers, thereby inducing the demands for OHM tokens. Stakers receive sOHM once they stake OHM tokens, and the number of sOHM tokens increases according to APY, which is a yield given by a smart contract. Finally, stakers receive the same amount of OHM tokens as sOHM when they unstake at a later time.

B. ANOMALY DETECTION USING DEEP LEARNING

Anomaly detection refers to the process of detecting an anomaly or outlier, which is a type of data representing patterns deviating from the normal category. Anomaly detection has been actively researched over the past few decades

in several fields including the detection and monitoring of financial misdeeds such as the fraudulent use of credit cards, financial transaction fraud, market manipulation, cyber security, quality management, medical and healthcare risks, and smart manufacturing [2], [3].

An anomaly can be categorized into various types such as point, conditional, and group anomalies [4], [32], [33]. A point anomaly is an individual data point or sequence having unusual values compared to other data. Examples include an excessively high financial transactions or abnormal health indicators of a patient's health. A conditional anomaly refers to an individual data point or sequence having unusual values in a specific context. For example, a sharp decline in temperature during the summer in seasonal temperature data is a conditional anomaly. A group anomaly refers to a group of data demonstrating unusual patterns compared to other groups of data. Here, an individual data point belonging to group anomaly can be normal. Examples of group anomaly include transactions demonstrating strange patterns in continuous financial transactions or repetitions of normal system logs at specific times.

Deep learning technology is being applied in numerous fields because of the rapid developments in the artificial intelligence field. Anomaly detection is an important issue that is actively researched in various domains, and therefore, many studies have focused on using deep learning technology in the anomaly detection domain. For example, in [3], deep anomaly detection is categorized into the following three types: (1) deep learning for feature extraction, (2) end-to-end anomaly score learning, and (3) learning feature representations of normality. In (1) deep learning and anomaly detection are separated completely where deep learning is used only for feature extraction. Research belonging to this type utilizes deep learning technology for extracting low-dimensional feature representations from high-dimensional data. These studies applied and implemented pre-trained deep learning models; however, completely separating feature extraction and anomaly scoring can interfere with deducing optimal results [34], [35], [36]. In (2), deep learning and anomaly scoring module are integrated completely, wherein a neural network that learns anomaly scores in an end-to-end method is used. Research in this type focuses on simultaneously learning feature representation and anomaly scores. Because previous anomaly measures are not used, loss functions with excellent performance must be designed for anomaly score learning [37], [38], [39]. In (3), deep learning and anomaly scoring module are not completely separated, and thus, it aims to learn normality effectively. Studies on autoencoders [8], [40], [41], generative adversarial networks [42], [43], [44], and predictability modeling [45], [46], [47] belong to this category.

Time-series anomaly detection refers to detecting abnormal data, or anomaly (point anomaly, conditional anomaly, and group anomaly), in time-series data arranged in a chronological order among various types of data. Data are constantly generated in real time in many application fields; thus, detecting and responding to anomaly in early stages by monitoring the time-series data are significant for heightening the efficiency and safety of a relevant domain. Therefore, time-series anomaly detection is extensively researched in a variety of domains including finance, medical, environment, and manufacturing process. Most studies on time-series anomaly detection using deep learning technology belong to learning feature representations of normality among the different categories of deep anomaly detection. With respect to anomaly detection methods, research on time-series anomaly detection can be categorized into reconstruction, forecasting, and dissimilarity methods [2], [3], [4], [5].

Reconstruction methods involve reconstructing data and detecting anomalies using the difference between original and reconstructed data. Autoencoder (AE) [40], [48], [49], VAE [8], [9], [10], [11], [12], [50], and transformer based models [5], [6], [7], [51], [52] use such reconstruction errors. In [50], a semi-supervised framework is introduced, employing a VAE and a one-class support vector machine for the detection of structural anomalies. In [51], a transformer-based generative adversarial network(GAN) framework is presented for time series anomaly detection. Reference [52] proposes an adversarial transformer model designed for detecting anomalies in multivariate time series data. Forecasting methods involve predicting the future state based on past and present states and detecting anomaly using the difference between the predicted and observed values [45], [53], [54]. Dissimilarity methods detect an anomaly by measuring the dissimilarity of data distribution or the distance from clusters where similar data are clustered [55], [56], [57].

VAE-LSTM [8] uses both a VAE module for identifying local features of a short window and an LSTM module for estimating the general correlation in the long term. An encoder in the VAE module generates low-dimensional embedding for a short window, the LSTM receives the generated embedding as an input and predicts the next embedding, and the decoder receives the predicted embedding as an input and reconstructs the original window. The reconstruction error is the anomaly detection score of VAE-LSTM. Anomaly transformer [5] presents a new anomaly attention module that simultaneously calculates prior association focusing on nearby data and series association that identifies association from the perspective of entire data. Prior association applies the learnable Gaussian kernel, while series association functions similar to the self-attention of a general transformer. In addition, the anomaly transformer amplifies the difference between normal and abnormal data by applying minimax association learning.

Anomaly detection still has numerous challenges to overcome because of its distinctive characteristics. Unlike other common problems, anomaly detection occurs very rarely and cannot be predicted [3]. Most datasets have a significantly greater amount of normal data and extremely small amounts of anomalies. Due to this class imbalance, it is challenging

or even impossible to obtain labeled data. Further, normal data can be misrecognized as anomaly due to noise, which is a type of error that can irregularly occur in the process of collecting and processing data. In addition, anomaly is frequently associated with factors that cannot be known in advance such as financial fraud or cyberattacks. Anomaly is defined differently and has varying characteristics for each domain, and therefore, a general anomaly detection model cannot be applied identically to different domains.

The blockchain technology has recently been under the spotlight; however, this technology has not matured and there is a completely different technological difficulty arising from the nature of a decentralized peer-to-peer network. Thus, research on anomaly detection in a blockchain network using deep learning technology is still in its early stages [1], [2], [3], [4], [5], [6], [7], [8], [9], [10], [11], [12], [13], [14], [15].

For blockchain networks, research is more actively conducted on a specific type of fraud, or a Ponzi scheme, which has narrower scope than anomaly detection [16], [17], [18], [19], [20], [21], [22], [23], [24], [25], [26], [27], [28], [29]. A Ponzi scheme is a traditional financial investment fraud that lures users by guaranteeing high profits. A Ponzi scheme on a blockchain network can induce more serious damages because of the anonymity of a blockchain and the unchangeable and unstoppable execution characteristics of a smart contract [16], [17].

In [16], smart contracts having the features of a Ponzi scheme were detected and analyzed to examine the dataset of 184 Ponzi schemes. The results of analyzing inflow and outflow transaction, life span, volume of payment, and payment inequality of 184 Ponzi schemes are presented. In [18], features were extracted from the opcode of user accounts and smart contracts, and then, a Ponzi scheme was detected using data mining and machine learning methods. In [21], 172 Ponzi schemes and 3,203 non-Ponzi smart contracts were used as a dataset, and a Ponzi detection model based on data mining to which opcode feature and behavior-based features are applied was proposed.

In [23], exploit transactions and attacker EOAs were analyzed for understanding attacks toward decentralized applications (Dapps) on Ethereum, and DEFIER, which is a tool for investigating new Dapps attacks, was proposed. In [24], several features such as the time difference between the first and last transactions, entire Ether balance, and minimum value of the received Ether were used, and a model was proposed to detect illicit accounts on the Ethereum network based on transaction history.

DeFi is growing continuously as it continues to provide innovative financial services based on a blockchain. DeFi is overcoming the limitations of traditional financial systems and substantially contributing to heightening the accessibility and transparency of financial services. However, abnormal transactions and fraudulent practices are present in the DeFi ecosystem, and there is a lack of research on anomaly detection in DeFi, whereas research on the overall security of DeFi is still in the rudimentary stages. Thus, research on anomaly detection in DeFi is inevitable for protecting users and enhancing the safety and reliability of the DeFi ecosystem.

FIGURE 1. Overall process of anomaly detection in Olympus DAO.

III. PROPOSED MODEL

This study proposes an anomaly VAE-Transformer model that is newly designed for anomaly detection in DeFi. This study targeted Olympus DAO, which is a popular DeFi protocol, and the proposed model can be applied to other DeFi protocols through minor modifications. Fig. 1 shows the process of anomaly detection in Olympus DAO. Data related to Olympus DAO are collected from on-chain data saved in the Ethereum blockchain, and then, they are analyzed to extract appropriate features. In the subsequent step, such data are used to train the proposed anomaly VAE-Transformer model. Finally, the trained model is used to detect anomaly in Olympus DAO, and the detection results are analyzed.

In the time-series data analysis, data are aggregated by the specific time interval (minute, hour, day, etc.) to be used. Hourly data are the most appropriate for Defi analyses. The frequency of data generation is insufficient to conduct the analysis on a minute basis considering the time required for the blockchain consensus or the transaction period of users. Daily data cannot properly reflect the high volatility of the DeFi market, and therefore, large fluctuations that occur for a day can be missed. The proposed model utilizes the hourly data for anomaly detection; the method for collecting the appropriate data based on understanding Olympus DAO and extracting appropriate features to constitute hourly data is explained in Section III-A.

For performing anomaly detection in DeFi, various types of data must be analyzed and high-dimensional data need to be examined because of the complexity of DeFi. Although hourly data are examined in this study, fairly long time sequence (e.g., one month) data need to be inspected instead of the data of a few hours to capture long-term dependency between data and to increate detection accuracy. Considering these circumstances, we propose an approach to combine the VAE and the transformer. Unlike traditional RNNs such as LSTM, the transformer has an outstanding capability to process long-term dependency in long-sequence data because of the self-attention mechanism. However, the transformer entails a high computational cost when utilizing extremely long sequences or high-dimensional data because it requires quadratic computational complexity. Further, VAE reduces the complexity of high-dimensional data and projects data onto a low-dimensional latent space for better representation.

The proposed anomaly VAE-Transformer model detects anomaly in a high-dimensional long data sequence based on the integration between the VAE and the transformer. The proposed model uses VAE to encode the sequence into low-dimensional embedding and the transformer to capture long-term dependency among embeddings in the embedding sequences. The proposed model uses reconstructions errors of both the VAE and the transformer for anomaly detection. The reconstruction error of the transformer reflects the extent of data anomaly from the long-term perspective, whereas the reconstruction error of VAE reflects the extent of data anomaly from the short-term perspective. The detailed architecture of the proposed method as well as the training and anomaly detection methods are provided in Section III-B.

A. DATA COLLECTION AND FEATURE EXTRACTION

Using diverse types of data plays a crucial role in anomaly detection and improving detection accuracy. This principle applies to anomaly detection targeting Olympus DAO. Data diversity helps detect anomalies in different scenarios or patterns; however, using an excessive amount of different data types can also lead to problems. Excessive data diversity causes overfitting of the detection model and lowers the detection accuracy for new data. High-dimensional data induce the curse of dimensionality, which increases the computational amount and reduces the performance of the model. If data types are too diverse, the quality of a specific data type may be too poor or inconsistent, and this will lower the accuracy of a model. Thus, an appropriate level of data diversity should be maintained by selecting highly relevant data for anomaly detection.

The types of data related to Olympus DAO include treasury balance, OHM price, yield (APY), and OHM market capitalization. We focused on the flow of OHM tokens coming into Olympus DAO from outside or those taken out externally as indicated by the solid red lines in Fig. 2. The proposed model monitors the following events and detects anomalies in Olympus DAO: an event where an external OHM token comes into Olympus DAO because of a user's staking, an event where an OHM token owned by Olympus is taken out externally because of unstaking, an event where a new OHM token is minted because of bond creation, and an event where the OHM token is sent to external users because of bond redemption.

Treasury balance and OHM market capitalization are closely related to staking and bonding activities, and APY is changed based on the pre-determined policy according to the total OHM supply. Therefore, closely observing the flow of OHM tokens enables changes in the Olympus DAO state to be identified, which in return, help detect anomalies. The OHM price is associated with the status of Olympus DAO; however, it is insufficient to detect an anomaly in Olympus DAO because the OHM price is highly volatile depending on the overall cryptocurrency market atmosphere. Transactions and transfer activities between OHM token holders are not related to anomalies in Olympus DAO.

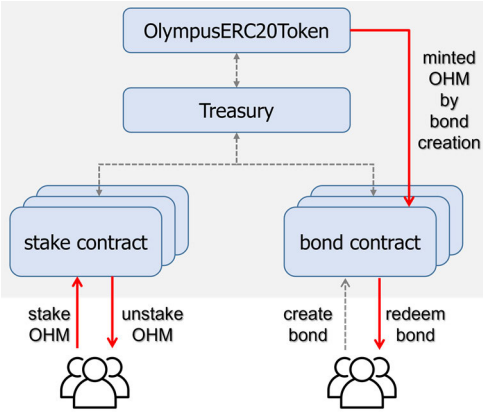

FIGURE 2. Flow of OHM tokens based on user activity (solid red lines). In addition to the simplified architecture of Olympus DAO, the flow of incoming/outgoing OHM tokens attributed to staking, unstaking, bond creation, and bond redemption is shown in this figure. In the case of bond creation, a new OHM token is minted internally.

We collected the users' staking, unstaking, bond creation, and bond redemption activities; these activities are performed when transactions generated and propagated by users are triggered. These transactions can be collected easily through transparency and integrity, which are advantages of a blockchain. However, unlike previous research on fraud and Ponzi schemes in Bitcoin and Ethereum, it is difficult to grasp the specific meaning of activities for DeFi such as Olympus DAO based only on the information of collected transactions. A user's transaction becomes a starting point in DeFi protocols. However, numerous complicated smart contracts interact simultaneously, which requires a deep understanding of the relevant DeFi protocol. Furthermore, DeFi platforms generate and use their own custom tokens such as OHM tokens, and the transfer information of these tokens is not specified in transactions unlike how the amount of transferred ETH is specified in the transactions. Therefore, the users' transactions, internal transactions among smart contracts invoked accordingly, and event logs generated as the result were collected and analyzed to identify the specific meaning of each action.

Table 1 summarizes the number of transactions collected and analyzed by activity and number of related smart contracts. Olympus DAO has a relatively shorter active period; however, the several occurrences of updates and changes took place during which new smart contracts were generated and used. Therefore, there are 3 versions of smart contracts for staking, and 18 types of smart contracts for bonding. We analyzed smart contract codes and the transactions sent to these contracts for obtaining the OHM tokens transferred by each transaction. In this process, the list of transactions was obtained with the help of Etherscan, which is an Ethereum block explorer, and Infura was used to obtain the details of transactions and corresponding receipts to be analyzed. Algorithm 1 shows the pseudocode for obtaining redemption activity information of OHM / DAI Bond V4. A total of

TABLE 1. Summary of collected and analyzed transactions.

Category	# of contracts	# of transactions
Staking	3	279551
Unstaking	3	132870
Bond creation	18	8669
Bond redemption	18	38361
Total	21	459451

Algorithm 1 Obtaining Redemption Activity Information of OHM/DAI Bond V4

Input: *addr* ▷ contract address of OHM/DAI Bond V4
Output: R ▷ set of (transaction, OHM amount) pairs
/ * get transactions of the contract from Etherscan */
TxList ← GetTxsFromAddress (*addr*)
$R \leftarrow \emptyset$
for all $tx \in TxList$ **do**
  if *tx.s tatus*= error **then** continue
  end if
  if *tx.m ethod* is *redeem* **then**
    /∗ get receipt of *tx* via Infura * /
    receipt ← GetTxReceipt (*tx*)
    for all *event* ∈ *receipt.logs* **do**
      if *event.name is BondRedeemed* **then**
        /* get OHM token amount inevent * /
        amount ← GetOHMAmount (*event*)
        $R \leftarrow R \cup \{(tx, amount)\}$
      end if
    end for
  end if
end for
return R

459,451 transactions were analyzed, and the detailed information is provided in Appendix.

We collected and analyzed the data generated from March 2021 to December 2022. Among them, the data from April 2021 were used for anomaly detection because the Olympus DAO was started in March 2021 and the activity patterns in the very beginning differed from the normal patterns of a later time. Only data up to April 2022 were used for the anomaly detection because bonding, which is one of the core ideas of Olympus DAO, was terminated at the end of April 2022 after its last transaction. Then, inverse bonding with an opposite concept of previous bonding was introduced, and it demonstrated completely different patterns from previous bonding activities. Therefore, we decided to exclude the data after May 2022.

These time-series data were resampled by hour to extract features to be used in the anomaly detection model. The following features were extracted every hour for each staking, unstaking, bond creation, and bond redemption activity, and a total of 12 features were obtained.

- Amount of transferred OHM tokens: The amount of OHM tokens transferred from Olympus DAO to outside or from outside into Olympus DAO in an hour.
- Number of transactions: The number of transactions generated and normally executed by users' activities in an hour.
- Number of active users: The number of active users who submitted new transactions in an hour is obtained by the number of unique addresses of the senders of transactions generated during this period.

B. ANOMALY VAE-TRANSFORMER

1) OVERALL ARCHITECTURE

The proposed anomaly VAE-Transformer model combines VAE for extracting local information in the short term and a transformer for identifying the dependency between data in the long term. Fig. 3 shows the overall architecture of the proposed model. The VAE model consists of an encoder and a decoder. The VAE encoder encodes daily data (sequence of hourly data of 24 hours) among time series data into low-dimensional embedding, and the VAE decoder receives the output of the transformer as an input to reconstruct the daily data. The transformer was designed by referring to the standard transformer suggested in [31] and the informer suggested in [58]; it consists of three encoders which use stacked attention and feed-forward layers. The transformer receives the encoding value of q non-overlapping daily data, and it generates q contextualized embeddings as an output, which is then transferred to the VAE decoder. The proposed model calculates the anomaly score using the difference between the original and reconstructed data, and it is determined as an anomaly if the anomaly score exceeds the threshold.

For time series $\mathrm{X} = \{x_1, x_2, \ldots, x_N\}$, $x_t \in \mathbb{R}^m$ represents m-dimensional data observed at time t. In this study, x_t represents hourly data at time t for m features of the DeFi protocol extracted in Section III-A. Fig. 4 shows that this study applies the overlapped sliding window technique to time-series X and generates daily data, or the sequence of hourly data of 24 h (window size $p = 24$), to be used as an input. A total of $(N - p + 1)$ daily data are generated from N hourly data, and the daily data at time t is expressed as $d_t = [x_t, x_{t+1}, \ldots, x_{t+p-1}]$. Therefore, x_i, which is the hourly data at time i, is overlapped with p number of daily data, which needs to be considered when calculating the anomaly score at time i. (However, for x_i *where* $1 \leq i < por\ (N - p) < i \leq N$, the number of overlapped daily data is less than p.)

2) TRAINING THE ANOMALY VAE-TRANSFORMER

The proposed anomaly VAE-Transformer model is trained in an unsupervised method in which VAE is trained first followed by the transformer, which is trained using the previously trained VAE.

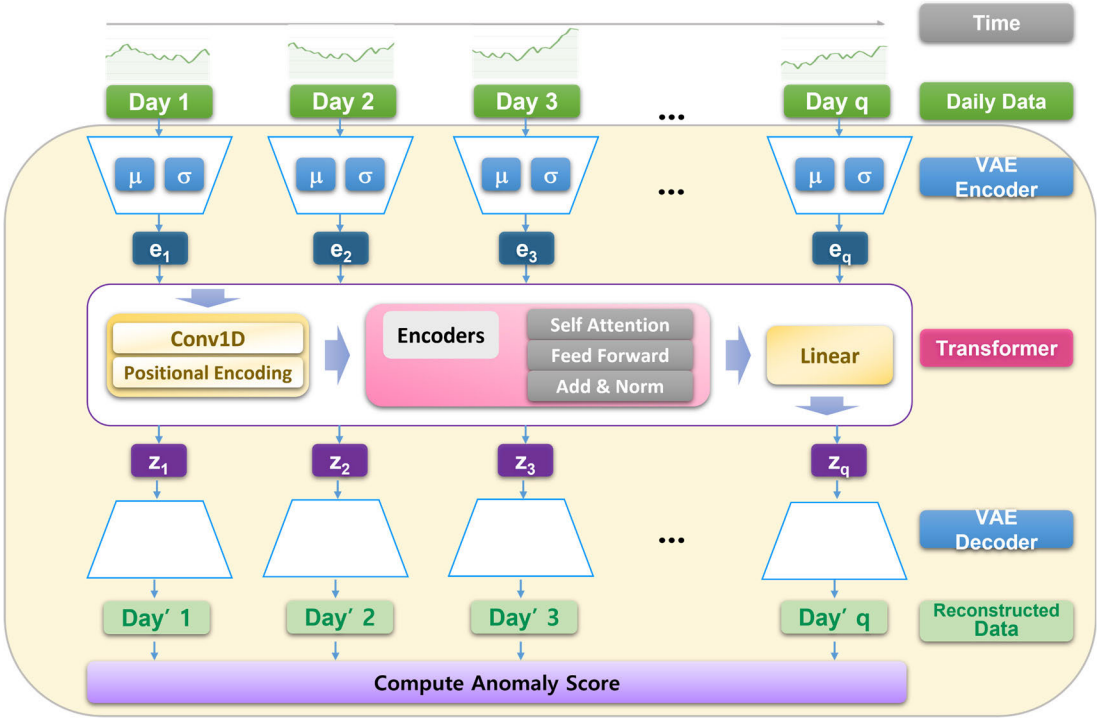

FIGURE 3. Overall architecture of anomaly VAE-Transformer model.

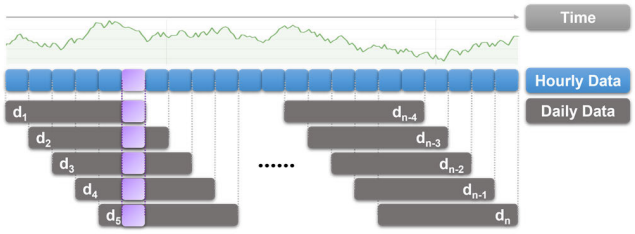

FIGURE 4. Generating daily data from the sequence of hourly data.

For training the VAE, the VAE encoder receives $d_t = [x_t, x_{t+1}, \ldots, x_{t+p-1}]$ *where* $d_t \in \mathbb{R}^{p\times m}$, which is the daily data at time t, and it encodes this data into low-dimensional embedding e_t.

$$e_t = VAE\_Encoder(d_t), \tag{1}$$

where $e_t \in \mathbb{R}^k$ and k represents the latent space dimension. The VAE decoder receives the VAE encoder's output e_t as an input, and it decodes this into reconstructed daily data $\hat{d}_t$.

$$\hat{d}_t = VAE\_Decoder(e_t), \tag{2}$$

where $\hat{d}_t \in \mathbb{R}^{p\times m}$. For minimizing the reconstruction error or the difference between the original daily data d_t and reconstructed daily data $\hat{d}_t$, VAE is optimized using ELBO loss. For VAE training, $(N - p + 1)$ number of overlapped daily data generated by time-series $X = \{x_1, x_2, \ldots, x_N\}$ are used.

For training, the transformer takes E_t which is the value encoded from q non-overlapping daily data D_t by the VAE encoder as an input, and it generates q number of contextualized embeddings as output Z_t.

$$D_t = \left[d_t, d_{t+p}, d_{t+p\times 2}, \ldots, d_{t+p\times(q-1)}\right], \tag{3}$$

$$E_t = \left[e_t, e_{t+p}, e_{t+p\times 2}, \ldots, e_{t+p\times(q-1)}\right], \tag{4}$$

$$\begin{aligned} Z_t &= Transformer\ (E_t) \\ &= \left[z_t^t, z_{t+p}^t, z_{t+p\times 2}^t, \ldots, z_{t+p\times(q-1)}^t\right], \end{aligned} \tag{5}$$

where $D_t \in \mathbb{R}^{q\times p\times m}$, $E_t \in \mathbb{R}^{q\times k}$, $z_i^t \in \mathbb{R}^k$, $Z_t \in \mathbb{R}^{q\times k}$, and the transformer is optimized to minimize the reconstruction loss of E_t and Z_t. In this training, all sequences of the q non-overlapped daily data generated from time-series X are used.

3) ANOMALY SCORE

After training, the proposed model can detect anomalies. When time-series $X = \{x_1, x_2, \ldots, x_N\}$ where $x_t \in \mathbb{R}^m$ is given, the anomaly score S_i for x_i at time t is calculated, and it is detected as an anomaly if this value exceeds the threshold θ.

For the given time-series X, the daily data at time t is expressed as $d_t = [x_t, x_{t+1}, \ldots, x_{t+p-1}]$, $d_t \in \mathbb{R}^{p\times m}$. In our model, D_t, the sequence of q non-overlapping daily data starting at time t is taken as an input.

$$D_t = \left[d_t, d_{t+p}, d_{t+p\times 2}, \ldots, d_{t+p\times(q-1)}\right], \tag{6}$$

where $D_t \in \mathbb{R}^{q\times p\times m}$. The VAE encoder encodes q non-overlapping daily data separately, and delivers E_t, which is

a set of q low-dimensional embeddings to the transformer.

$$e_t = VAE\_Encoder(d_t), \tag{7}$$
$$E_t = \left[e_t, e_{t+p}, e_{t+p\times 2}, \ldots, e_{t+p\times(q-1)}\right], \tag{8}$$

where $e_t \in \mathbb{R}^k$, $E_t \in \mathbb{R}^{q\times k}$.

The transformer receives E_t as an input and generates q contextualized embeddings as a set of output Z_t, which is then delivered to the VAE decoder.

$$Z_t = Transformer\ (E_t) = \left[z_t^t, z_{t+p}^t, z_{t+p\times 2}^t, \ldots, z_{t+p\times(q-1)}^t\right], \tag{9}$$

where $Z_t \in \mathbb{R}^{q\times k}$, $z_i^t \in \mathbb{R}^k$.

The VAE decoder receives the transformer output Z_t as an input, and decodes q number of z_i^t separately; as a result, O_t, a set of q reconstructed daily data are generated.

$$o_i^t = VAE\_Decoder\ \left(z_i^t\right) = \left[\hat{x}_i^t, \hat{x}_{i+1}^t, \ldots, \hat{x}_{i+p-1}^t\right], \tag{10}$$
$$O_t = \left[o_t^t, o_{t+p}^t, o_{t+p\times 2}^t, \ldots, o_{t+p\times(q-1)}^t\right], \tag{11}$$

where $o_i^t \in \mathbb{R}^{p\times m}$, $O_t \in \mathbb{R}^{q\times p\times m}$. Here, o_i^t represents the reconstructed daily data of d_i, which is the daily data at time i among the sequence $D_t = \left[d_t, d_{t+p}, d_{t+p\times 2}, \ldots, d_{t+p\times(q-1)}\right]$, which started at time t. $\hat{x}_i^t$ represents the reconstructed hourly data at time i.

$LossH_i^t$ is calculated using the mean squared error (MSE) of the original hourly data x_i and reconstructed hourly data $\hat{x}_i^t$.

$$LossH_i^t = MSE\left(x_i, \hat{x}_i^t\right). \tag{12}$$

As explained above, since each hourly data x_i is duplicated in the p daily data, the anomaly score S_i at time i is obtained by calculating the average of $LossH_i^t$ where $(i-p+1) \le t \le i$.

$$S_i = Average\left(LossH_i^t\right) = \frac{1}{p}\sum_{t=i-p+1}^{i} LossH_i^t. \tag{13}$$

Fig. 5 shows the process of calculating the anomaly score; if this score exceeds a certain threshold, x_i is judged as an anomaly. All steps of finding anomalies are depicted in Algorithm 2.

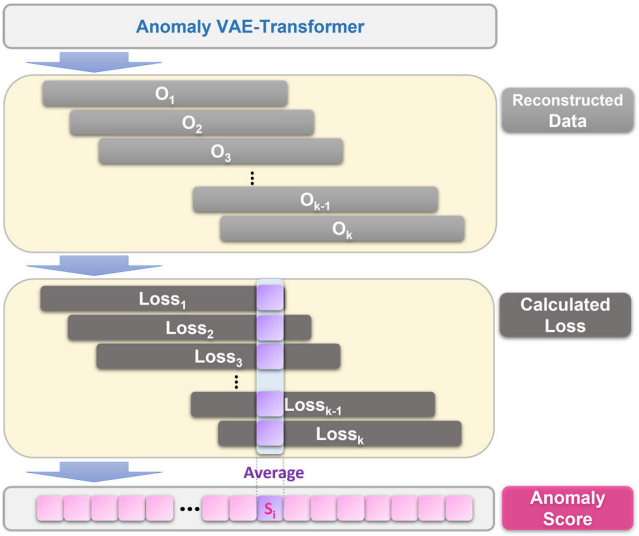

FIGURE 5. Process of computing the anomaly score from the output of the anomaly VAE-Transformer.

Algorithm 2 Finding Anomaly in Time Series

Input: X ▷ time series $X = \{x_1, x_2, \cdots, x_N\}$
Output: *Anomaly* ▷ set of detected anomalies

Anomaly $\leftarrow \emptyset$
for $t = 1$ to $N - pq + 1$ **do**
  /* compose a sequence of daily data
    where $d_t = \left[x_t, x_{t+1}, \cdots, x_{t+p-1}\right]$*/
  $D_t \leftarrow \left[d_t, d_{t+p}, \cdots, d_{t+p\times(q-1)}\right]$
  /* get VAE embedding*/
  $E_t \leftarrow$ VAE_Encoder (D_t)
  /* get Transformer output*/
  $Z_t \leftarrow$ Transformer (E_t)
  /* get reconstructed sequence of daily data
  $O_t = \left[o_t^t, o_{t+p}^t, \cdots, o_{t+p\times(q-1)}^t\right]$
    where $o_i^t = \left[\hat{x}_i^t, \hat{x}_{i+1}^t, \cdots, \hat{x}_{i+p-1}^t\right]$*/
  $O_t \leftarrow$ VAE_Decoder (Z_t)
  /* calluilate reconnstruction error */
  $LossH_i^t \leftarrow \text{MSE}\left(x_i, \hat{x}_i^t\right) \quad \forall i, t \le i < t + pq$
end for
for $i = 1$ to N **do**
  /* calculate anomaly score at timei*/
  $S_i \leftarrow \frac{1}{p}\sum_{t=i-p+1}^{i} \text{Loss}\, H_i^t$
  if $S_i > threshold$ **then**
    $Anomaly \leftarrow Anomaly \cup \{x_i\}$
  end if
end for
return *Anomaly*

IV. EXPERIMENTS AND RESULTS

A. IMPLEMENTATION AND EXPERIMENTS

The dataset used in the experiment and analysis was generated through data collection and feature extraction, and it consists of hourly data from 00:00 on April 1, 2021 to 24:00 on April 30, 2022. Each data has 12 features, and the total number of data is 9,480. This dataset was divided into two datasets; the first one with 3,600 data until 24:00 on August 28, 2021 was used for training, while the remaining 5,880 data were used for inference and analysis.

Unsupervised learning is generally applied for anomaly detection. Abnormal data are extremely rare compared to the number of normal data, and therefore, it is extremely challenging to obtain enough labeled abnormal data. Furthermore, the patterns of anomaly vary significantly, and it is difficult to identify future patterns from the past patterns. Thus, unsupervised learning is suitable for learning the patterns of normal data and for detecting the patterns of

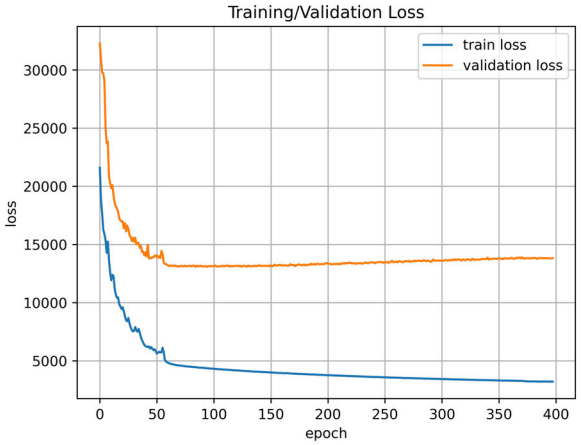

FIGURE 6. Training loss and validation loss of VAE (dimension of latent space = 64, size of hidden layers = [512, 512]).

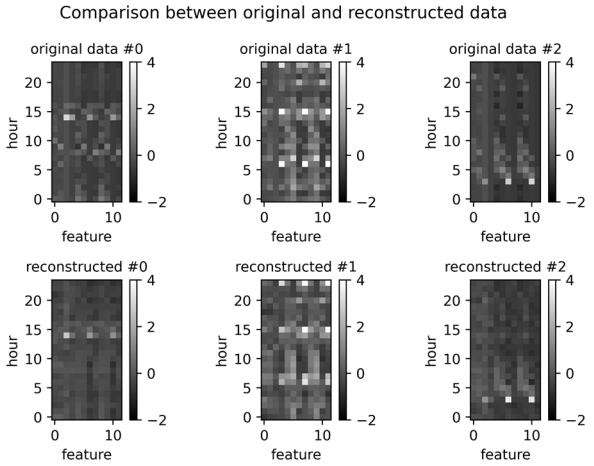

FIGURE 7. Visualized comparison between the original and VAE-reconstructed data. Three arbitrary data are selected where the reconstructed data (bottom) is highly similar to the original data (top).

anomaly. We adopted unsupervised learning, and the training process of the proposed anomaly VAE-Transformer model was divided into 1) VAE training and 2) transformer training using the trained VAE.

VAE is a generative model that can infer the generation factors of training data and provide excellent anomaly detection results for data within a short window. The VAE of the proposed model is implemented using the base version of TimeVAE [11], which is appropriate for handling the time-series data. We used the VAE to encode the sequence of hourly data of a day (24 h) into low-dimensional embedding, and we performed decoding to reconstruct the data of one day from the transformer output into daily data. The training method of VAE is as follows.

The window size is set to 24 and 3,577 input windows are generated from 3,600 training data using the sliding window technique. After shuffling these windows, 20% of the windows are used for validation, and the windows are optimized using the ELBO loss. Grid search is employed to find the optimized values for the dimension of a latent space and the number of hidden layers in VAE. The grid for the dimension of a latent space is defined as [2, 4, 8, 16, 32, 64, 128, 256, 512], and the grid for the number of layers is defined as [1], [2], and [3]. Subsequently, after generating all possible combinations of these values, each combination is applied to train the model, and the results are evaluated. The experiment is conducted by varying the dimension of a latent space and the number of hidden layers in VAE; specifically, the most outstanding performance is obtained when the dimension of the latent space is 64 and the number of hidden layers is 2. Fig. 6 shows the relevant losses, and Fig. 7 shows the visualization result of arbitrarily selecting three input windows and inputting them in the trained VAE to generate reconstructed output. The reconstructed output is highly similar to the original data.

The transformer can help sequential data processing, and it has been widely useful for finding the long-range temporal dependency of time-series data. The transformer part of the proposed model is implemented based on the informer [58] and anomaly transformer [5] using the standard transformer provided by PyTorch. In the proposed model, the data of 28 days (approx. one month) are input in the transformer, and the input data are composed of embeddings in the unit of days generated by the pre-trained VAE encoder. The transformer reconstructs the data of 28 days from this input and delivers the data to the subsequent step.

We proceeded with the training as explained below to ensure that the transformer adequately reconstructs the input. A total of 2,929 windows with a size of 672 were generated using the sliding window technique from 3,600 training data. Windows with a size of 28 were newly generated by gathering embeddings created by inputting in the VAE encoder for every 24 data in each window. 80% of the windows in the front were used for training, while 20% the back were used for validation. Training was performed to minimize the MSE of input data and reconstruct the data using Adam optimization. The training process is stopped within 100 epochs with a batch size of 16. Table 2 presents the hyperparameters used in the experiment.

TABLE 2. Hyperparameters of proposed Anomaly VAE-Transformer.

Sub-model	Hyperparameter	Value
VAE	Window size	24
	Number of features	12
	Dimension of latent space	64
	Hidden layer size	512
	Number of hidden layers	2
Transformer	Sequence length	28
	Dimension of representation	512
	Number of heads	8
	Dimension of feedforward network	512
	Number of encoder layers	3
	Dropout	0.2

TABLE 3. Elapsed time of each process in Anomaly VAE-Transformer.

Process	Performance factor	Time(ms)
VAE training	Average training time per given dimension and layers	304,829
	Average time elapsed per epoch	449
Transformer training	Total elapsed time	33,226
	Average time elapsed per epoch	1,263
VAE encoding	Total elapsed time	7,787
	Average time elapsed per window	2
Transformer encoding	Total elapsed time	2,186
	Average time elapsed per window	6
Reconstructing	Total elapsed time	392

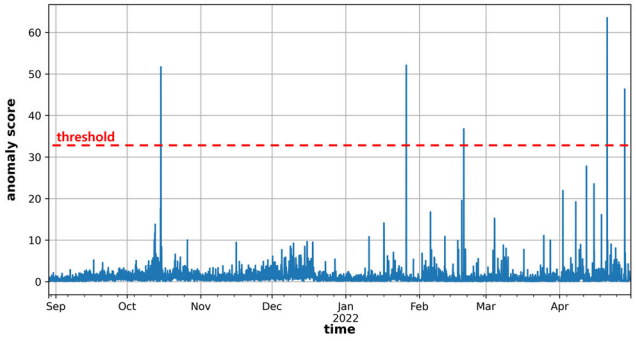

FIGURE 8. Anomaly score calculated for each time period (from 00:00 on August 29, 2021 to 24:00 on April 30, 2022; threshold = 32.3).

TABLE 4. Detected anomalies (in descending order of anomaly score).

Score rank	Period	Anomaly score
#1	2022-04-21 00:00–01:00	63.54
#2	2022-01-26 11:00–12:00	52.07
#3	2021-10-15 05:00–06:00	51.70
#4	2021-10-15 04:00–05:00	49.72
#5	2022-04-28 09:00–10:00	46.36
#6	2022-04-20 23:00–24:00	39.40
#7	2022-02-19 16:00–17:00	36.79

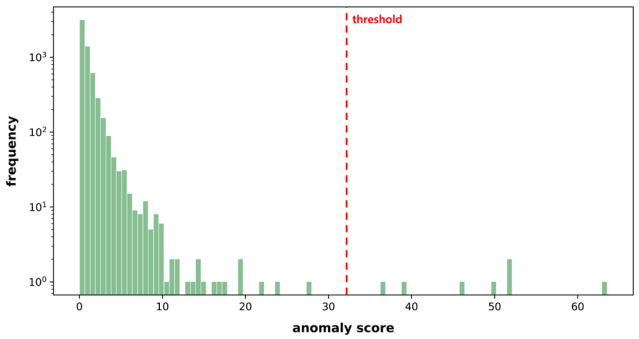

FIGURE 9. Log scale distribution of anomaly scores. Most normal data are clearly distinguished from certain anomaly data, and the threshold is 32.3.

The experiments were conducted on a machine equipped with Intel Core i7-12700F CPU and NVIDIA GeForce RTX 3060 GPU and 32GB DDR4 RAM. We measured the time required for training both VAE and transformer, as well as the time taken for calculating anomaly scores using the trained model. The results of system performance are summarized in Table 3.

B. ANALYSIS

Sometimes data without accurate labels or ground truth are analyzed in the studies on anomaly detection. In such cases, it is difficult to use traditional performance indicators such as F1 score or confusion matrix. This limitation becomes more prominent during anomaly detection in DeFi. In this case, the detection performance of a model is proved through various case studies to determine the suitability of the model. To the best of our knowledge, no anomaly detection case has been officially reported for Olympus DAO. Thus, this study aims to prove through case studies that the proposed anomaly VAE-Transformer can successfully detect various abnormal patterns in the Olympus DAO.

For anomaly detection, the anomaly score is calculated for every hour from 00:00 on August 29, 2021 to 24:00 on April 30, 2022 using the trained anomaly VAE-Transformer. Fig. 8 shows the anomaly score of each period, while Fig. 9 shows the log scale of the distribution of anomaly score values. Most periods have low scores; however, certain periods have abnormal scores, which indicate that an anomaly has occurred in those periods. We set the threshold θ for distinguishing between normal and abnormal scores as 32.3 to allow the periods with the top 0.1% scores to become an anomaly.

Table 4 presents the detected anomaly periods. If the continuous periods are combined (#1 and #6, #3 and #4), there are a total of five periods; the top four cases are analyzed to verify whether anomaly detection is properly executed.

Case 1: Sudden increase in user activity (2022-04-20 23:00–2022-04-21 01:00)

Fig. 10 shows the results of visualizing the changes in 12 features in the detected (marked with red) and adjacent periods. As shown in the figure, the staking, unstaking, and redemption activity of the OHM tokens varied significantly in the red-colored periods. In the case of staking and unstaking, the numbers of active users and transactions suddenly increased, which led to a significant increase in the token amount. In terms of bond redemption, only one transaction occurred in the period; however, a substantially large transaction amount of 59,000 OHM tokens was involved in the redemption process, which is very likely to be an abnormal transaction. Similarly, seven features showed significant changes from adjacent values, and thus, this period received high anomaly score and it was detected as anomaly. Unfortunately, the reason for such an abnormal activity in this period remains unknown; however, this period was a few days

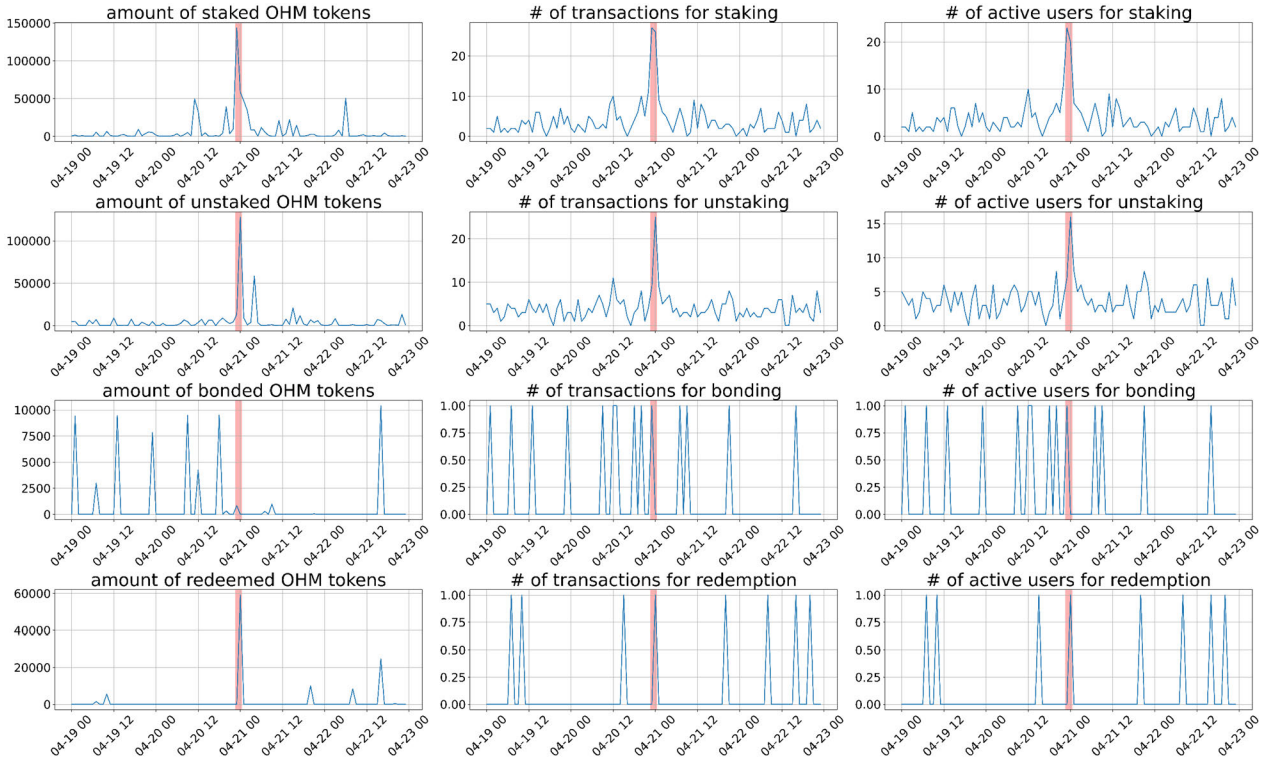

FIGURE 10. (Case 1) Graphs of 12 features in the detected anomaly period (2022-04-20 23:00∼2022-04-21 01:00) and adjacent periods. Red parts indicate anomaly. Features related to staking, unstaking, and bond redemption activities sharply increased.

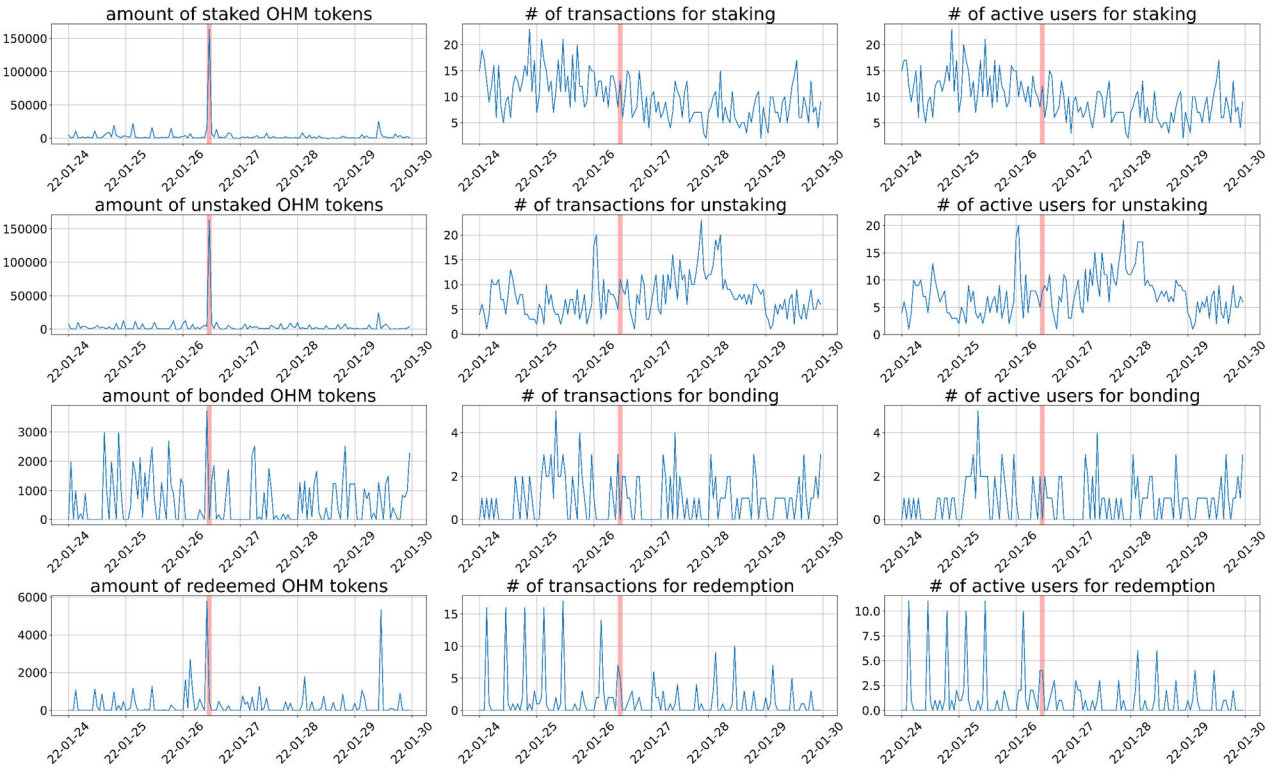

FIGURE 11. (Case 2) Graphs of 12 features in the detected anomaly period (2022-01-26 11:00 - 12:00) and adjacent periods. Red parts indicate anomaly. The amount of staked OHM tokens and unstaked OHM tokens significantly increased, but no noticeable changes occurred in other features.

before the inverse bond started, which is a notable change in Olympus DAO. The anticipation and anxiety stemming from recent price drop and introduction of new inverse bond are assumed to have caused the anomaly.

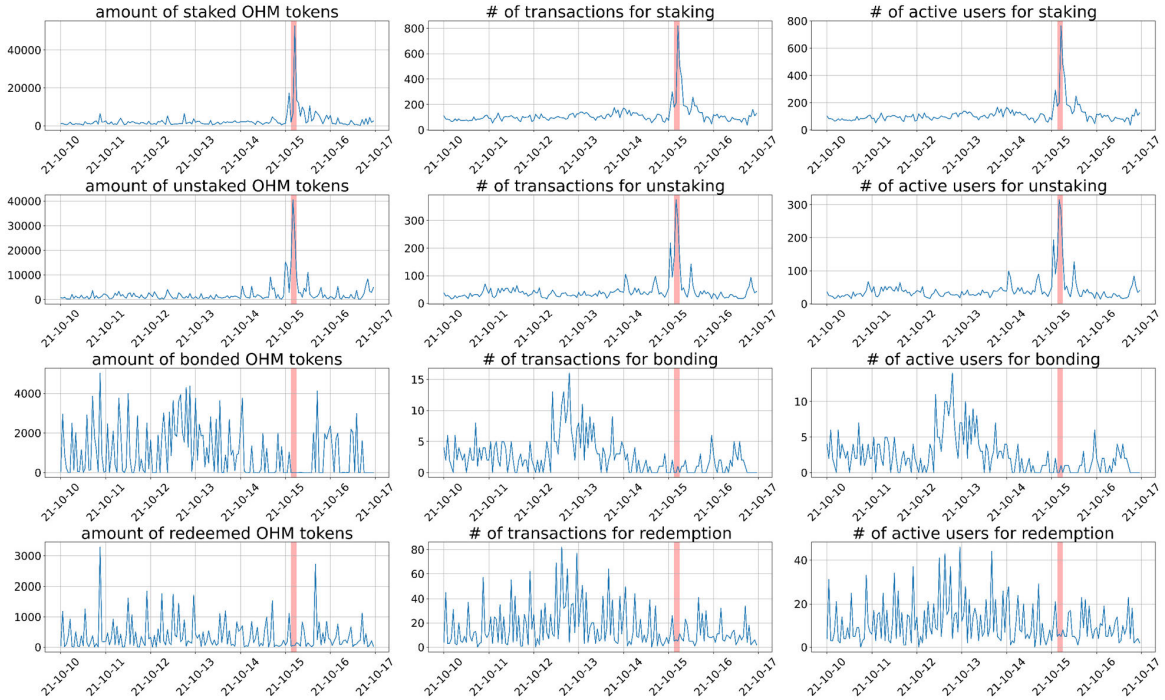

FIGURE 12. (Case 3) Graphs of 12 features in the detected anomaly period (2022-10-15 04:00–06:00) and adjacent periods. Red parts indicate anomaly. Staking and unstaking activities sharply increased in which the number of transactions and unique addresses (number of active users) is particularly high. In contrast, bond creation and redemption activities are extremely low.

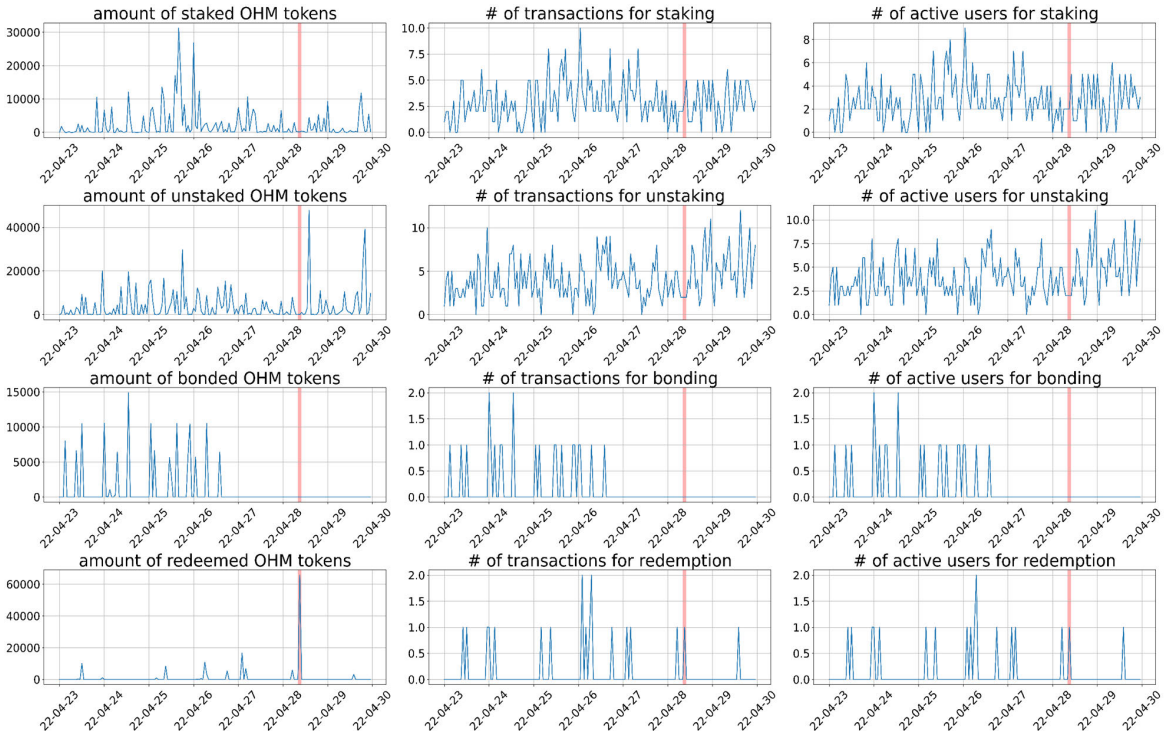

FIGURE 13. (Case 4) Graphs of 12 features in the detected anomaly period (2022-04-28 09:00–10:00) and the adjacent periods. Red parts indicate an anomaly. User activity is extremely low.

Case 2: Abnormal activity of a specific user (2022-01-26 11:00–12:00)

Fig. 11 shows that other features did not fluctuate significantly in this period; however, the amounts of staked OHM tokens and unstaked OHM tokens increased considerably compared to the adjacent values. Despite a large increase in the amount, the number of transactions and active users did not vary noticeably, and therefore, it can be inferred that a small number of users executed staking and unstaking for OHM tokens on a large scale in this period.

TABLE 5. Description of Olympus DAO transactions collected.

Category	Version	Type	Subversion	Smart contract address	# of transactions
Staking			V1	0x0822f3c03dcc24d200aff33493dc08d0e1f274a2	24658
			V2	0xc8c436271f9a6f10a5b80c8b8ed7d0e8f37a612d	215208
			V3	0xb63cac384247597756545b500253ff8e607a8020	39685
			Total		279551
Unstaking			V1	0x0822f3c03dcc24d200aff33493dc08d0e1f274a2	9392
			V2	0xfd31c7d00ca47653c6ce64af53c1571f9c36566a	85242
			V3	0xb63cac384247597756545b500253ff8e607a8020	38236
			Total		132870
Bond creation	V1	OHM / LUSD LP	V1	0xfb1776299e7804dd8016303df9c07a65c80f67b6	234
		OHM / FRAX LP	V1	0x539b6c906244ac34e348bbe77885cdfa994a3776	36
			V2	0xc20cfff07076858a7e642e396180ec390e5a02f7	556
			V3	0x99e9b0a9dc965361c2cbc07525ea591761aeaa53	75
		OHM / DAI LP	V1	0xd27001d1aaed5f002c722ad729de88a91239ff29	289
			V2	0x13e8484a86327f5882d1340ed0d7643a29548536	76
			V3	0x996668c46fc0b764afda88d83eb58afc933a1626	74
			V4	0x956c43998316b6a2f21f89a1539f73fb5b78c151	1102
		LUSD	V1	0x10c0f93f64e3c8d0a1b0f4b87d6155fd9e89d08d	552
		FRAX	V1	0x8510c8c2b6891e04864fa196693d44e6b6ec2514	540
			V2	0xc60a6656e08b62dd2644dc703d7855301363cc38	466
		ETH	V2	0xe6295201cd1ff13ced5f063a5421c39a1d236f1c	1239
		CVX	V1	0x6754c69fe02178f54ada19ebf1c5569826021920	22
			V2	0x767e3459a35419122e5f6274fb1223d75881e0a9	114
		DAI	V1	0xa64ed1b66cb2838ef2a198d8345c0ce6967a2a3c	245
			V2	0xd03056323b7a63e2095ae97fa1ad92e4820ff045	184
			V3	0x575409f8d77c12b05fed8b455815f0e54797381c	1943
	V2			0x9025046c6fb25fb39e720d97a8fd881ed69a1ef6	922
	Total				8669
Bond redemption	V1	OHM / LUSD LP	V1	0xfb1776299e7804dd8016303df9c07a65c80f67b6	1891
		OHM / FRAX LP	V1	0x539b6c906244ac34e348bbe77885cdfa994a3776	190
			V2	0xc20cfff07076858a7e642e396180ec390e5a02f7	3097
			V3	0x99e9b0a9dc965361c2cbc07525ea591761aeaa53	941
		OHM / DAI LP	V1	0xd27001d1aaed5f002c722ad729de88a91239ff29	1193
			V2	0x13e8484a86327f5882d1340ed0d7643a29548536	213
			V3	0x996668c46fc0b764afda88d83eb58afc933a1626	277
			V4	0x956c43998316b6a2f21f89a1539f73fb5b78c151	9838
		LUSD	V1	0x10c0f93f64e3c8d0a1b0f4b87d6155fd9e89d08d	1891
		FRAX	V1	0x8510c8c2b6891e04864fa196693d44e6b6ec2514	2475
			V2	0xc60a6656e08b62dd2644dc703d7855301363cc38	2106
		ETH	V2	0xe6295201cd1ff13ced5f063a5421c39a1d236f1c	3459
		CVX	V1	0x6754c69fe02178f54ada19ebf1c5569826021920	125
			V2	0x767e3459a35419122e5f6274fb1223d75881e0a9	714
		DAI	V1	0xa64ed1b66cb2838ef2a198d8345c0ce6967a2a3c	779
			V2	0xd03056323b7a63e2095ae97fa1ad92e4820ff045	748
			V3	0x575409f8d77c12b05fed8b455815f0e54797381c	7502
	V2			0x9025046c6fb25fb39e720d97a8fd881ed69a1ef6	922
	Total				38361

We thoroughly investigated staking and unstaking transactions that occurred in this period, and we discovered one suspicious transaction. A user with the Ethereum address 0 × 41339.9825963515 e5705df8d3b0ea98105ebb1c unstaked a large amount of 79844 OHM tokens and then immediately staked them again. After a few minutes, the same amount of OHM tokens was unstaked and immediately staked again. Repeatedly executing unstaking and staking within a short period of time does not allow users to gain any benefit and only causes losses from transaction fees, which raised suspicion behind their actions. Such an action may be part of a more complicated attack on DeFi which could be analyzed with the timestamp attack model in [59], or an attempt to find the vulnerability of the staking code, or a simple mistake of the user. The proposed model cannot identify the purpose of this action; however, it can detect the period in which

this action has a high probability of abnormal transactions. The effectiveness of this model was confirmed because the detection result induces further investigation for preventing more serious incidents.

Case 3: Structural changes in Olympus DAO (2021-10-15 04:00–06:00)

In this period, activities related to staking and unstaking significantly increased, while those related to bond creation and redemption greatly decreased, as shown in Fig. 12. A notable aspect is that the number of staking and unstaking transactions is extremely high; the value is the highest in the entire analysis dataset. This result indicates that the anomaly in this period was caused by a large number of normal users affected by a specific incident rather than by a small number of certain individuals. In this period, the Olympus DAO V2 to which new governance and bonding policy are applied was announced officially. The anticipation and anxiety for the new version and a sudden increase in the OHM price caused active staking and unstaking activities among normal users. Furthermore, previous bonding related activities were subsided because the significant changes in bonding were forecasted. As shown here, the proposed model can adequately detect sudden anomalies of normal users arising from changes in the Olympus DAO itself.

Case 4: Extremely low user activity (2022-04-28 09:00–10:00)

As shown in Fig. 13, this period has very low activities compared to those in other periods. The amount and transaction number of staking and unstaking activities were both lower than those of the adjacent values. For bond creation, no new bonding activities were observed since 43 h before this period. Unlike previous periods of active participation, low activities increased the anomaly score in this period. Further, one transaction with a noticeable amount of redemption raised the anomality of this period. This period experienced a transition where the existing bond mechanism ended and a new inverse bond mechanism started. Uncertainty about the future led users to restrain from participating in new activities. The redemption of existing tokens occurred in preparation for terminating existing bond service, and new bonding activities were not observed.

The analysis results showed that all four cases had a period in which an anomaly occurred, ultimately proving that the proposed model can accurately detect anomaly periods. In case 1, an anomaly period was successfully detected where multiple features demonstrated sudden changes; in case 2, the anomaly of a specific user caused by malicious activities or mistakes was detected. In case 3, anomaly was detected when numerous normal users intensively participated in activities because of the changes in the Olympus DAO. In contrast, in case 4, the anomaly of a stagnant state was detected in which the activities of normal users were decreased substantially. These analysis results confirmed that the proposed anomaly VAE-Transformer model successfully detected various anomaly patterns, and it is a suitable model for anomaly detection in the Olympus DAO.

V. CONCLUSION

This study proposed a new methodology for anomaly detection in DeFi. First, on-chain data of Olympus DAO, which is a popular DeFi protocol, were collected and analyzed to extract 12 features that can identify the flow of OHM tokens. In addition, we proposed a new deep learning model, the anomaly VAE-Transformer model, which consists of a transformer that captures dependency between data in the long term, and VAE, which extracts local information in the short term. This model was used to perform anomaly detection in the Olympus DAO based on an actual dataset, and the four anomaly cases with the highest anomaly scores in the detection results were analyzed further to prove the effectiveness of the proposed model.

DeFi has high technological complexity because of a complex protocol structure, interactions among various smart contracts, and diverse token transactions, which has been causing difficulty in performing anomaly detection. In particular, DeFi protocols such as Olympus DAO have extremely high volatility, which makes it difficult to monitor and manage potential risks. The proposed method overcomes such limitations and enables anomaly detection to be conducted effectively. The proposed method helps not only the governance of DeFi but also general users participating in DeFi. From the perspective of DeFi governance, anomaly detection can be considered an opportunity to find errors or the vulnerability of the DeFi protocol and to identify malicious attack attempts of an attacker. From the perspective of general users, anomaly detection can help promptly recognize important structural changes of the DeFi protocol they are participating in, changes in user trends, and sudden issues. These advantages can help the stakeholders make decisions based on relevant information. In addition, anomaly detection using the proposed method protects users' assets, raises the transparency of the DeFi market, and provides trust required for new investors and enterprises to participate in the DeFi ecosystem. Thus, this study is expected to contribute to promoting the development of the DeFi market.

The suggested method is applicable to Olympus DAO, but it can also be applied to other DeFi protocols through slight modifications. However, the expansion and application of the proposed method cannot be easily automated and require assistance of experts. The data collection and extraction of appropriate features is the most difficult to automate because DeFi protocols have their own unique concept and architecture. In future studies, latest machine learning techniques will be researched to overcome the abovementioned limitations, and new methods will be explored to easily automate the proposed method for various DeFi protocols.

APPENDIX

COLLECTED TRANSACTION DATA

In our study, we collected transaction data pertaining to user activities such as staking, unstaking, bond creation, and bond redemption to train and analyze the anomaly detection model. As can be observed in Table 5, these transactions amounted

to a total of 459,451 instances across 21 smart contracts. The fragmentation, as a result of supporting bonding with various cryptocurrencies like DAI, necessitated numerous smart contracts. Further, with every upgrade, due to the inherent nature of blockchain technology, new versions of smart contracts were deployed for use. Utilizing block explorers like Etherscan, transactions related to the specific smart contract addresses mentioned in Table 5 can be observed.

REFERENCES

[1] DeFiLlama. *DeFi Total Value Locked(TVL)*. Accessed: May 20, 2023. [Online]. Available: https://defillama.com/

[2] S. Schmidl, P. Wenig, and T. Papenbrock, "Anomaly detection in time series: A comprehensive evaluation," *Proc. VLDB Endowment*, vol. 15, no. 9, pp. 1779–1797, May 2022.

[3] G. Pang, C. Shen, L. Cao, and A. V. D. Hengel, "Deep learning for anomaly detection: A review," *ACM Comput. Surv.*, vol. 54, no. 2, pp. 1–38, Mar. 2021.

[4] K. Choi, J. Yi, C. Park, and S. Yoon, "Deep learning for anomaly detection in time-series data: Review, analysis, and guidelines," *IEEE Access*, vol. 9, pp. 120043–120065, 2021.

[5] J. Xu, H. Wu, J. Wang, and M. Long, "Anomaly transformer: Time series anomaly detection with association discrepancy," 2021, *arXiv:2110.02642*.

[6] Z. Chen, D. Chen, X. Zhang, Z. Yuan, and X. Cheng, "Learning graph structures with transformer for multivariate time-series anomaly detection in IoT," *IEEE Internet Things J.*, vol. 9, no. 12, pp. 9179–9189, Jun. 2021.

[7] X. Wang, D. Pi, X. Zhang, H. Liu, and C. Guo, "Variational transformer-based anomaly detection approach for multivariate time series," *Measurement*, vol. 191, Mar. 2022, Art. no. 110791.

[8] S. Lin, R. Clark, R. Birke, S. Schönborn, N. Trigoni, and S. Roberts, "Anomaly detection for time series using VAE-LSTM hybrid model," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2020, pp. 4322–4326.

[9] D. Park, Y. Hoshi, and C. C. Kemp, "A multimodal anomaly detector for robot-assisted feeding using an LSTM-based variational autoencoder," *IEEE Robot. Autom. Lett.*, vol. 3, no. 3, pp. 1544–1551, Jul. 2018.

[10] Z. Li, Y. Zhao, J. Han, Y. Su, R. Jiao, X. Wen, and D. Pei, "Multivariate time series anomaly detection and interpretation using hierarchical inter-metric and temporal embedding," in *Proc. 27th ACM SIGKDD Conf. Knowl. Discovery Data Mining*, Aug. 2021, pp. 3220–3230.

[11] A. Desai, C. Freeman, Z. Wang, and I. Beaver, "TimeVAE: A variational auto-encoder for multivariate time series generation," 2021, *arXiv:2111.08095*.

[12] X. Wang, Y. Du, S. Lin, P. Cui, Y. Shen, and Y. Yang, "adVAE: A self-adversarial variational autoencoder with Gaussian anomaly prior knowledge for anomaly detection," *Knowl.-Based Syst.*, vol. 190, Feb. 2020, Art. no. 105187.

[13] M. U. Hassan, M. H. Rehmani, and J. Chen, "Anomaly detection in blockchain networks: A comprehensive survey," *IEEE Commun. Surveys Tuts.*, vol. 25, no. 1, pp. 289–318, 1st Quart., 2023.

[14] T. Pham and S. Lee, "Anomaly detection in Bitcoin network using unsupervised learning methods," 2016, *arXiv:1611.03941*.

[15] C. Yan, C. Zhang, Z. Lu, Z. Wang, Y. Liu, and B. Liu, "Blockchain abnormal behavior awareness methods: A survey," *Cybersecurity*, vol. 5, no. 1, p. 5, Dec. 2022.

[16] M. Bartoletti, S. Carta, T. Cimoli, and R. Saia, "Dissecting Ponzi schemes on Ethereum: Identification, analysis, and impact," *Future Gener. Comput. Syst.*, vol. 102, pp. 259–277, Jan. 2020.

[17] Y. Zhang, W. Yu, Z. Li, S. Raza, and H. Cao, "Detecting Ethereum Ponzi schemes based on improved LightGBM algorithm," *IEEE Trans. Computat. Social Syst.*, vol. 9, no. 2, pp. 624–637, Apr. 2022.

[18] W. Chen, Z. Zheng, J. Cui, E. Ngai, P. Zheng, and Y. Zhou, "Detecting Ponzi schemes on Ethereum: Towards healthier blockchain technology," in *Proc. World Wide Web Conf.*, 2018, pp. 1409–1418.

[19] M. Bartoletti, B. Pes, and S. Serusi, "Data mining for detecting Bitcoin Ponzi schemes," in *Proc. Crypto Valley Conf. Blockchain Technol. (CVCBT)*, Jun. 2018, pp. 75–84.

[20] W. Chen, Z. Zheng, E. C.-H. Ngai, P. Zheng, and Y. Zhou, "Exploiting blockchain data to detect smart Ponzi schemes on Ethereum," *IEEE Access*, vol. 7, pp. 37575–37586, 2019.

[21] E. Jung, M. Le Tilly, A. Gehani, and Y. Ge, "Data mining-based Ethereum fraud detection," in *Proc. IEEE Int. Conf. Blockchain (Blockchain)*, Jul. 2019, pp. 266–273.

[22] K. Toyoda, T. Ohtsuki, and P. T. Mathiopoulos, "Identification of high yielding investment programs in Bitcoin via transactions pattern analysis," in *Proc. IEEE Global Commun. Conf.*, Dec. 2017, pp. 1–6.

[23] L. Su, X. Shen, X. Liao, X. F. Wang, and L. Xing, "Evil under the sun: Understanding and discovering attacks on Ethereum decentralized applications," in *Proc. USENIX Secur. Symp.*, 2021, pp. 1307–1324.

[24] S. Farrugia, J. Ellul, and G. Azzopardi, "Detection of illicit accounts over the Ethereum blockchain," *Exp. Syst. Appl.*, vol. 150, Jul. 2020, Art. no. 113318.

[25] M. Ostapowicz and K. Żbikowski, "Detecting fraudulent accounts on blockchain: A supervised approach," in *Web Information Systems Engineering—WISE 2019*. Hong Kong: Springer, Jan. 2019, pp. 18–31.

[26] M. Bhowmik, T. S. S. Chandana, and B. Rudra, "Comparative study of machine learning algorithms for fraud detection in blockchain," in *Proc. 5th Int. Conf. Comput. Methodologies Commun. (ICCMC)*, Apr. 2021, pp. 539–541.

[27] P. N. Sureshbhai, P. Bhattacharya, and S. Tanwar, "KaRuNa: A blockchain-based sentiment analysis framework for fraud cryptocurrency schemes," in *Proc. IEEE Int. Conf. Commun. Workshops (ICC Workshops)*, Jun. 2020, pp. 1–6.

[28] N. Dhieb, H. Ghazzai, H. Besbes, and Y. Massoud, "A secure AI-driven architecture for automated insurance systems: Fraud detection and risk measurement," *IEEE Access*, vol. 8, pp. 58546–58558, 2020.

[29] C. Lee, S. Maharjan, K. Ko, and J. W.-K. Hong, "Toward detecting illegal transactions on Bitcoin using machine-learning methods," in *Blockchain and Trustworthy Systems*. Guangzhou, China: Springer, Dec. 2020, pp. 520–533.

[30] D. P. Kingma and M. Welling, "Auto-encoding variational Bayes," 2013, *arXiv:1312.6114*.

[31] A. Vaswani, N. Shazeer, N. Parmar, and J. Uszkoreit, "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 1–11.

[32] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection: A survey," *ACM Comput. Surv.*, vol. 41, no. 3, pp. 1–58, Jul. 2009.

[33] A. Blázquez-García, A. Conde, U. Mori, and J. A. Lozano, "A review on outlier/anomaly detection in time series data," *ACM Comput. Surveys*, vol. 54, no. 3, pp. 1–33, Apr. 2022.

[34] W. Yu, W. Cheng, C. C. Aggarwal, K. Zhang, H. Chen, and W. Wang, "NetWalk: A flexible deep embedding approach for anomaly detection in dynamic networks," in *Proc. 24th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Jul. 2018, pp. 2672–2681.

[35] D. Xu, E. Ricci, Y. Yan, J. Song, and N. Sebe, "Learning deep representations of appearance and motion for anomalous event detection," 2015, *arXiv:1510.01553*.

[36] R. T. Ionescu, F. S. Khan, M.-I. Georgescu, and L. Shao, "Object-centric auto-encoders and dummy anomalies for abnormal event detection in video," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 7834–7843.

[37] W. Sultani, C. Chen, and M. Shah, "Real-world anomaly detection in surveillance videos," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 6479–6488.

[38] G. Pang, C. Yan, C. Shen, A. van den Hengel, and X. Bai, "Self-trained deep ordinal regression for end-to-end video anomaly detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 12170–12179.

[39] M.-H. Oh and G. Iyengar, "Sequential anomaly detection using inverse reinforcement learning," in *Proc. 25th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Jul. 2019, pp. 1480–1490.

[40] C. Zhang, "A deep neural network for unsupervised anomaly detection and diagnosis in multivariate time series data," in *Proc. AAAI. Artif. Intell.*, vol. 33, no. 1, 2019, pp. 1409–1416.

[41] W. Lu, Y. Cheng, C. Xiao, S. Chang, S. Huang, B. Liang, and T. Huang, "Unsupervised sequential outlier detection with deep architectures," *IEEE Trans. Image Process.*, vol. 26, no. 9, pp. 4321–4330, Sep. 2017.

[42] T. Schlegl, P. Seeböck, S. M. Waldstein, U. Schmidt-Erfurth, and G. Langs, "Unsupervised anomaly detection with generative adversarial networks to guide marker discovery," in *Information Processing in Medical Imaging*. Boone, NC, USA: Springer, Jun. 2017, pp. 146–157.

[43] J. Donahue, P. Krähenbühl, and T. Darrell, "Adversarial feature learning," 2016, *arXiv:1605.09782*.
[44] X. Xia, X. Pan, N. Li, X. He, L. Ma, X. Zhang, and N. Ding, "GAN-based anomaly detection: A review," *Neurocomputing*, vol. 493, pp. 497–535, Jul. 2022.
[45] M. Munir, S. A. Siddiqui, A. Dengel, and S. Ahmed, "DeepAnT: A deep learning approach for unsupervised anomaly detection in time series," *IEEE Access*, vol. 7, pp. 1991–2005, 2019.
[46] W. Liu, W. Luo, D. Lian, and S. Gao, "Future frame prediction for anomaly detection—A new baseline," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 6536–6545.
[47] M. Ye, X. Peng, W. Gan, W. Wu, and Y. Qiao, "AnoPCN: Video anomaly detection via deep predictive coding network," in *Proc. 27th ACM Int. Conf. Multimedia*, Oct. 2019, pp. 1805–1813.
[48] J. Audibert, P. Michiardi, F. Guyard, S. Marti, and M. A. Zuluaga, "USAD: Unsupervised anomaly detection on multivariate time series," in *Proc. 26th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2020, pp. 3395–3404.
[49] P. Zheng, S. Yuan, X. Wu, J. Li, and A. Lu, "One-class adversarial nets for fraud detection," in *Proc. AAAI Conf. Artif. Intell.*, vol. 33, no. 1, 2019, pp. 1286–1293.
[50] A. Pollastro, G. Testa, A. Bilotta, and R. Prevete, "Semi-supervised detection of structural damage using variational autoencoder and a one-class support vector machine," *IEEE Access*, vol. 11, pp. 67098–67112, 2023.
[51] A.-H. Shin, S. T. Kim, and G.-M. Park, "Time series anomaly detection using transformer-based GAN with two-step masking," *IEEE Access*, vol. 11, pp. 74035–74047, 2023.
[52] F. Zeng, M. Chen, C. Qian, Y. Wang, Y. Zhou, and W. Tang, "Multivariate time series anomaly detection with adversarial transformer architecture in the Internet of Things," *Future Gener. Comput. Syst.*, vol. 144, pp. 244–255, Jul. 2023.
[53] P. Malhotra, L. Vig, G. Shroff, and P. Agarwal, "Long short term memory networks for anomaly detection in time series," in *Proc. ESANN*, 2015, p. 89.
[54] H. Zhao, Y. Wang, J. Duan, C. Huang, D. Cao, Y. Tong, B. Xu, J. Bai, J. Tong, and Q. Zhang, "Multivariate time-series anomaly detection via graph attention network," in *Proc. IEEE Int. Conf. Data Mining (ICDM)*, Nov. 2020, pp. 841–850.
[55] L. Shen, L. Zhuocong, and J. Kwok, "Timeseries anomaly detection using temporal hierarchical one-class network," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 33, 2020, pp. 13016–13026.
[56] H. Song, Z. Jiang, A. Men, and B. Yang, "A hybrid semi-supervised anomaly detection model for high-dimensional data," *Comput. Intell. Neurosci.*, vol. 2017, pp. 1–9, Mar. 2017.
[57] J. Liu, H. Zhu, Y. Liu, H. Wu, Y. Lan, and X. Zhang, "Anomaly detection for time series using temporal convolutional networks and Gaussian mixture model," *J. Phys., Conf. Ser.*, vol. 1187, no. 4, Apr. 2019, Art. no. 042111.
[58] H. Zhou, S. Zhang, J. Peng, S. Zhang, J. Li, H. Xiong, and W. Zhang, "Informer: Beyond efficient transformer for long sequence time-series forecasting," in *Proc. AAAI Conf. Artif. Intell.*, vol. 35, no. 12, 2021, pp. 11106–11115.
[59] X. Yang, S. Wang, F. Li, Y. Zhang, W. Yan, F. Gai, B. Yu, L. Feng, Q. Gao, and Y. Li, "Ubiquitous verification in centralized ledger database," in *Proc. IEEE 38th Int. Conf. Data Eng. (ICDE)*, May 2022, pp. 1808–1821.

AHYUN SONG received the M.S. degree in computer science from KAIST, South Korea, in 2005. She is currently pursuing the Ph.D. degree in computer science and engineering with Sungkyunkwan University. From 2005 to 2015, she was the Manager with the Korea Financial Telecommunications and Clearings Institute. Since 2015, she has been a Senior Manager with the Financial Security Institute, South Korea. Her major research interests include security, blockchain, and DeFi.

EUISEONG SEO (Member, IEEE) received the B.S., M.S., and Ph.D. degrees in computer science from the Korea Advanced Institute of Science and Technology (KAIST), in 2000, 2002, and 2007, respectively. He is currently a Professor with the Department of Computer Science and Engineering, Sungkyunkwan University, South Korea. Before joining Sungkyunkwan University in 2012, he was an Assistant Professor with the Ulsan National Institute of Science and Technology (UNIST), South Korea, from 2009 to 2012, and a Research Associate with Pennsylvania State University, from 2007 to 2009. His research interests are system software, embedded systems, and cloud computing.

HEEYOUL KIM received the B.E., M.S., and Ph.D. degrees in computer science from KAIST, South Korea, in 2000, 2002, and 2007, respectively. From 2007 to 2008, he was with Samsung Electronics as a Senior Engineer. Since 2009, he has been a Faculty Member with the Department of Computer Science, Kyonggi University. His major research interests include cryptography, security, and blockchain.