

# Nilkamal School of Mathematics, Applied Statistics & Analytics, NMIMS

**MSc. Statistics and Data Science (2024-26)**

TITLE – DEFI ANONALY DETECTION

PREPARED BY –

GROUP-11

|  |  |  |  |
| --- | --- | --- | --- |
| NAME | SAP ID | ROLL NO | YEAR |
| Tanishtha Poddar | 86062400074 | A057 | 2024-26 |
| Ritika Pradhan | 86062400068 | A058 | 2024-26 |
| Reetik Prajapati | 86062400005 | A059 | 2024-26 |
| Verni Sharma | 86062300020 | A060 | 2024-26 |
| Gourish Salgaonkar | 86062400002 | A061 | 2024-26 |

UNDER THE SUPERVISION OF

Dr. Kavita Jain

|  |  |
| --- | --- |
| **Table of Contents** | |
| **Sr. No.** | **Topic** | |
| 1. | ACKNOWLEDGMENT | |
| 2. | ABSTRACT | |
| 3. | INTRODUCTION | |
| 4. | LITERATURE REVIEW | |
| 5. | AIMS AND OBJECTIVES | |
| 6. | ANORMALY AND FRAUD DETECTION | |
| 7. | MODEL DESCRIPTION | |
| 8. | EXPLORATORY DATA ANALYSIS Data Analysis (EDA) | |
| 9. | METHODOLOGY | |
| 10. | LIMITATIONS | |
| 11. | FUTURE SCOPE | |
| 12. | REFERENCES | |

**I. ACKNOWLEDGMENT**

Words on paper are mere ink marks, but when they serve a purpose, they carry the weight of thought and intention. We express our heartfelt gratitude to all those individuals whose guidance and support made the successful completion of this project possible.

We would like to sincerely thank our project guide, Dr. Kavita Jain, whose insightful guidance and unwavering support were instrumental in shaping and completing this project. Her direction and encouragement have been invaluable throughout.

We are also deeply grateful to Dr. Pradnya Khandeparkar, Ma’am, Programme Co-ordinator of M.Sc. Statistics and Data Science for her continuous support and motivation throughout all phases of the project.

Our sincere thanks extend to the entire Department of Statistics for their guidance and assistance at various stages, and for fostering an environment conducive to research and learning.

With utmost respect and appreciation, we acknowledge all the professors and staff members of NSoMASA for their valuable suggestions and for providing essential information and support. We are also thankful to our Dean, Dr. Sushil Kulkarni, for his encouragement.

Finally, this project would not have been possible without the collaborative efforts of our team members and the excellent mentorship we received. Their dedication and teamwork ensured the project was completed within the stipulated time**.**

**II. Abstract**

Blockchains' smart contracts have pioneered decentralized finance (DeFi) ecosystems that currently hold more than $160 billion in Total Value Locked (TVL). DeFi is as much known for its promise of high-yielding opportunities as for its vulnerabilities like protocol exploits, flash loan attacks, and more sophisticated financial fraud. This paper introduces a scalable anomaly detection system to detect malicious behavior in DeFi protocols through the examination of behavioral and transactional features across chains. Utilizing a new labeled transaction dataset covering 23 DeFi protocols (collectively obtained from Covalent), we utilize state-of-the-art machine learning techniques—Deep Neural Networks (DNNs), XGBoost, and a domain-specialized fine-tuned Large Language Model (LLM)—to identify anomalies. We design 414 DeFi-focused features that encode cross-chain behavioral patterns, greatly enhancing classification accuracy compared to traditional methods. Rooted in foundational blockchain innovations such as Bitcoin [1] and Ethereum [2], this research fills gaps in decentralized identity systems [4], smart media engagement [5], and DeFi protocol mechanics [6]–[9], providing a solid foundation for anomaly detection at scale.

**III. INTRODUCTION**

Bitcoin [1] first brought about peer-to-peer digital value transfer, while Ethereum [2] increased blockchain's applicability with programmable smart contracts to support decentralized applications (dApps). These concepts have since been adopted across various domains, ranging from IoT marketplaces [3] to decentralized identity systems [4] and media platforms [5]. Among them, DeFi has turned out to be a revolutionary phenomenon, providing permissionless financial services. However, its rapid growth has introduced systemic risks, including flash loan exploits, rug pulls, and money laundering [6]–[9]. These threats often manifest as anomalous transactional patterns, necessitating advanced detection mechanisms to ensure ecosystem integrity.

Past blockchain fraud detection work has been conducted in a specialized manner for isolated-chain analyses (e.g., Bitcoin or Ethereum) employing algorithms such as address clustering [4] or the detection of Ponzi schemes [8]. These approaches, however, do not account for cross-chain and cross-protocol relations characteristic of today's DeFi. Additionally, traditional models lack the ability to learn from and respond to changing behavioral patterns prevalent in decentralized ecosystems, e.g., collateralized debt loops [9] or governance token exploits [6].

This paper fills these gaps by introducing a multi-chain anomaly detection framework specifically designed for DeFi. Our contributions are:

* Labeled Dataset: A new dataset of transactions from 23 DeFi protocols on 12 chains, labeled with fraud tags (e.g., flash loans, rug pulls).
* Feature Engineering: 414 domain-aware features that reflect cross-chain liquidity flows, time-based transaction dependencies, and protocol interaction patterns.
* ML Framework: Comparative assessment of DNNs, XGBoost, and fine-tuned LLMs, which are tuned for class imbalance common in DeFi fraud cases.

**IV. LITERATURE REVIEW**

anomaly detection in real-time finance and Blockchain is gaining attention owing to the complications of the fraudulent activities. Nakamoto's Bitcoin [1] realized peer-to-peer digital value transfer, while Ethereum [2] extended this advantage to decentralized infrastructural smart contracts, together leading to innovations in DeFi.

Palaiokrassas et al. [3],[5] have examined the Blockchain usages in sensor data marketplaces and immersive media systems, which set the foundation for the behavior analysis of decentralized systems. Simultaneously, Liu et al. [4] documented a thorough view of Blockchain-based identity systems and stressed the need for entity-level modeling that is essential for anomaly detection tasks.

Recent works by Bartoletti et al. [6] and Kyriazis et al. [8] provide investigations into the vulnerabilities of DeFi lending pools and monetary interactions in the DeFi realm. Such findings warrant enhancing the need for anomaly detection mechanisms with the capability to perform globally over varied DeFi protocols and financial instruments.

Bouraga [8] studied the social sentiment amidst Blockchain forks' phenomenon and advanced the contention that external data—such as social signals—may signal impending anomalous events. Darlin et al. [9] demonstrated a systemic risk emanating from debt-financed collateral in DeFi, further propelling the notion of anomaly detection being a preemptive monitoring tool.

Finally, Trozze et al. [10] offered an overlap between investigative instruments and 'on-chain' analysis to detect money laundering and fraud schemes, highlighting the need for interdisciplinary approaches to assist practical DeFi surveillance."

**V. AIMS AND OBJECTIVES**

**5.1 Contextual Background**

The recent surge in growth of Decentralized Finance (DeFi) platforms has brought on unrivaled prospects for financial innovation, but at the same time, has enhanced exposure to high-level fraud, exploitation, and abnormal activities. DeFi platforms are unlike other traditional finance systems because they work on many blockchain networks (e.g., Ethereum, Binance Smart Chain, Solana) with their respective transactional structures and data schemas. Traditional anomaly detection approaches mostly work in isolated silos and are not interoperable and scalable to meet cross-chain challenges. This project fills these voids by developing a common, multi-blockchain anomaly and fraud detection platform. Through the use of hybrid approaches, the solution seeks to improve transparency, security, and trust in decentralized finance while evolving with the dynamic nature of cross-chain transactions.

* 1. **Aims**

1. **Design Scalable Anomaly Detection Framework on Multi-Chain:** Develop cross-chain interoperable pipeline for analysis of transactional data to be robustly adaptable to evolving DeFi protocols.
2. **Enhance Proactive Risk Mitigation in DeFi Ecosystems:** Enable real-time identification of high-risk activities such as flash loan attacks, rug pulls, wash trading, and smart contract exploits to minimize financial losses.
3. **Fill the Gap Between Rule-Based and Machine Learning Approaches**: Combine domain-specific heuristics with sophisticated ML models to enhance detection rates while ensuring interpretability for stakeholders.
   1. **Objectives**
4. **Cross-Chain Data Integration and Normalization:** 
   1. Collect and normalize transactional data (e.g., token exchanges, liquidity pool transactions, wallet addresses) across various blockchains into a common schema.
   2. Apply temporal and spatial feature engineering to identify anomalies in transaction frequency, volume, and network activity.
5. **Hybrid Detection Model Development:** 
   1. Integrate rule-based heuristics (threshold triggers for abrupt liquidity withdrawals) with unsupervised ML algorithms (Isolation Forest, LOF) to flag outliers.
   2. Train models to identify context-specific anomalies, e.g., pump-and-dump schemes or anomalous token price fluctuations on decentralized exchanges (DEXs).
6. **Real-Time Monitoring and Alert System:** 
   1. Implement a horizontally scalable pipeline for real-time transaction analysis using distributed computing frameworks (Apache Kafka, Spark) for low-latency processing.
   2. Create an API-based dashboard to present risk scores, suspicious transactions, and threat patterns to auditors and operators of DeFi platforms.
7. **Validation and Cross-Platform Interoperability:** 
   1. Compare model performance to benchmark against historical sets of attacks (e.g., Poly Network exploit, Cream Finance flash loans) to measure precision and recall.
   2. Make the integration compatible with top blockchain networks through modular design for easy integration with platforms such as Uniswap, PancakeSwap, and Aave.
8. **Stakeholder-Centric Risk Reporting:**
   1. Produce actionable insights and audit trails to facilitate regulatory compliance and decentralized autonomous organizations (DAOs) decision-making.
   2. Release open-source detection rules and model architectures to promote collaboration for the DeFi security community.

**VI. Anomaly and Fraud Detection in Decentralized Finance (DeFi)**

**6.1 Anomaly Detection:**

Anomaly detection in DeFi means detecting deviations from set transactional or behavioral patterns within blockchain networks. These deviations can be caused by innocuous factors (e.g., protocol updates, market fluctuations) or can be indicative of nascent risks (e.g., initial-stage exploits). Anomalies are defined by statistical outliers in transaction volume, frequency, or network interaction metrics. For example, an unexpected surge of token trades on a DEX or unscheduled liquidity pool withdrawals would signal abnormalities.

In multi-chain ecosystems, anomaly detection involves processing heterogeneous streams of data (e.g., Ethereum, Solana, Binance Smart Chain) to separate platform idiosyncrasies from systemic attacks. The ultimate purpose is to give warning signals early, allowing stakeholders to examine and fine-tune detection mechanisms in response to changing DeFi dynamics.

**6.2 Fraud Detection**

Fraud detection targets the detection of intentional, malicious behavior aimed at profiting from DeFi protocols. In contrast to anomalies, fraud entails willful deception, e.g., rug pulls (project abandonment after raising investments), flash loan attacks (manipulating asset prices through uncollateralized loans), or smart contract exploits (e.g., reentrancy attacks). Detection is based on following adversarial intent through transactional forensics, e.g., fund flow patterns to mixers or frequent malicious address interactions. Fraud detection platforms emphasize actionable information—like freezing fraudulent transactions or alerting stakeholders—to reduce losses. In cross-chain scenarios, this means correlating threats between blockchains to deal with coordinated attacks (e.g., multi-DEX arbitrage exploits).

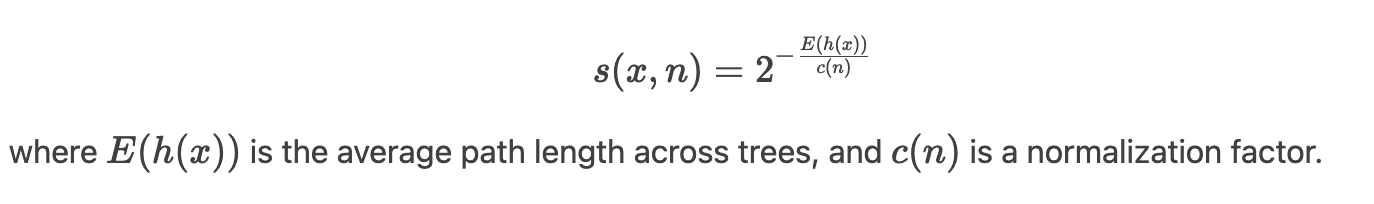
**VII. MODEL DESCRIPTION**

1. **Isolation Forest:**

Isolation Forest is an unsupervised anomaly detection algorithm that uses the concept of isolating anomalies by random partitioning. It builds an ensemble of binary decision trees (isolation trees) by recursively partitioning the data using randomly chosen features and split values. Anomalies, being rare and unique, are isolated near the root of the trees, and fewer splits are needed. The anomaly score is computed from the mean path length over all trees—shorter paths correspond to greater likelihood of anomalies. In contrast to density-based algorithms, Isolation Forest performs well on high-dimensional data and scales linearly with large volumes of transactions because of its linear time complexity.

Mathematical Foundation: For a data set of ‘*n’* instances, each isolation tree is constructed by selecting randomly a feature and split value until instances become isolated.

The anomaly score *'s'* for a point *'x'* is calculated as:

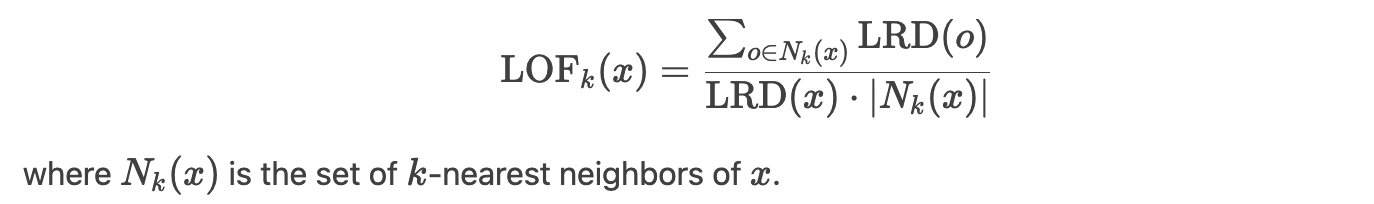


1. **Local Outlier Factor (LOF):**

LOF is a density-based unsupervised algorithm that measures how much a data point is isolated compared to its neighbors. It calculates local reachability density (LRD) for every point, which gives the inverse of the average reachability distance to its *'k'* k-nearest neighbors. The LOF value is the ratio of the LRD of a point to the average LRD of its neighbors. Points with LOF values much larger than 1 are marked as outliers.

Key Elements: Reachability Distance: The maximum of the real distance between two points and the *'k'* k-distance (distance to the *'k'* k-th neighbor).

LOF Formula:

****

1. **DBSCAN (Density-Based Spatial Clustering):**

DBSCAN discovers clusters as denser regions of data points by sparse areas of points (noise). DBSCAN specifies:

* 1. Core Points: Points with at least MinPts neighbors in distance ϵ.
  2. Border Points: Points within ϵ of a core point but not having MinPts neighbors.
  3. Noise: Points that are not core nor border.

The algorithm expands clusters incrementally from core points, which makes it stable to arbitrary shapes of clusters. Anomalies are defined as noise points lying in low-density areas.

Parameters:

ϵ: Neighborhood defining radius.

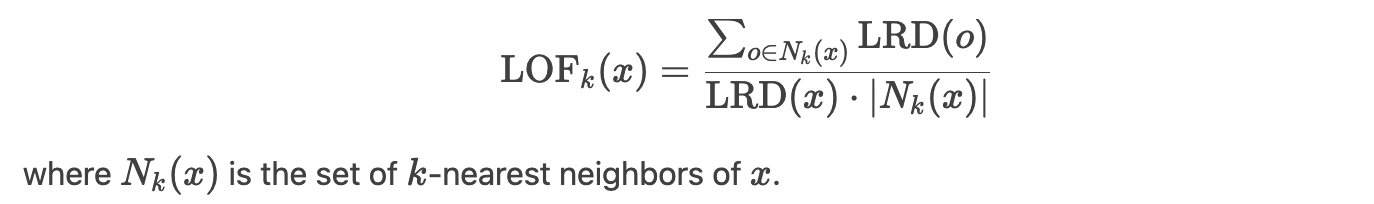
MinPts: Number of minimum neighbors to create a dense area.

1. **Variational Autoencoder (VAE):**

A generative neural network that learns a probabilistic latent representation of input data. It has:

* 1. Encoder: Projects input data to a latent space distribution (mean μ and variance σ).
  2. Decoder: Reconstructs data from samples drawn from the latent distribution.

Anomalies are identified by monitoring the reconstruction error—high-error data points (badly reconstructed) are marked. The VAE minimizes the evidence lower bound (ELBO) loss, trading off reconstruction quality and latent space regularization.



1. **LSTM (Long Short-Term Memory):**

A form of recurrent neural network (RNN) that is used to model temporal sequences. It employs memory cells and gating mechanisms (input, forget, output gates) to hold long-term dependencies. For anomaly detection, the LSTM is trained on normal time-series data (e.g., transaction intervals). Anomalies are detected by deviations from predicted sequences (high prediction error).

Gating Mechanisms:

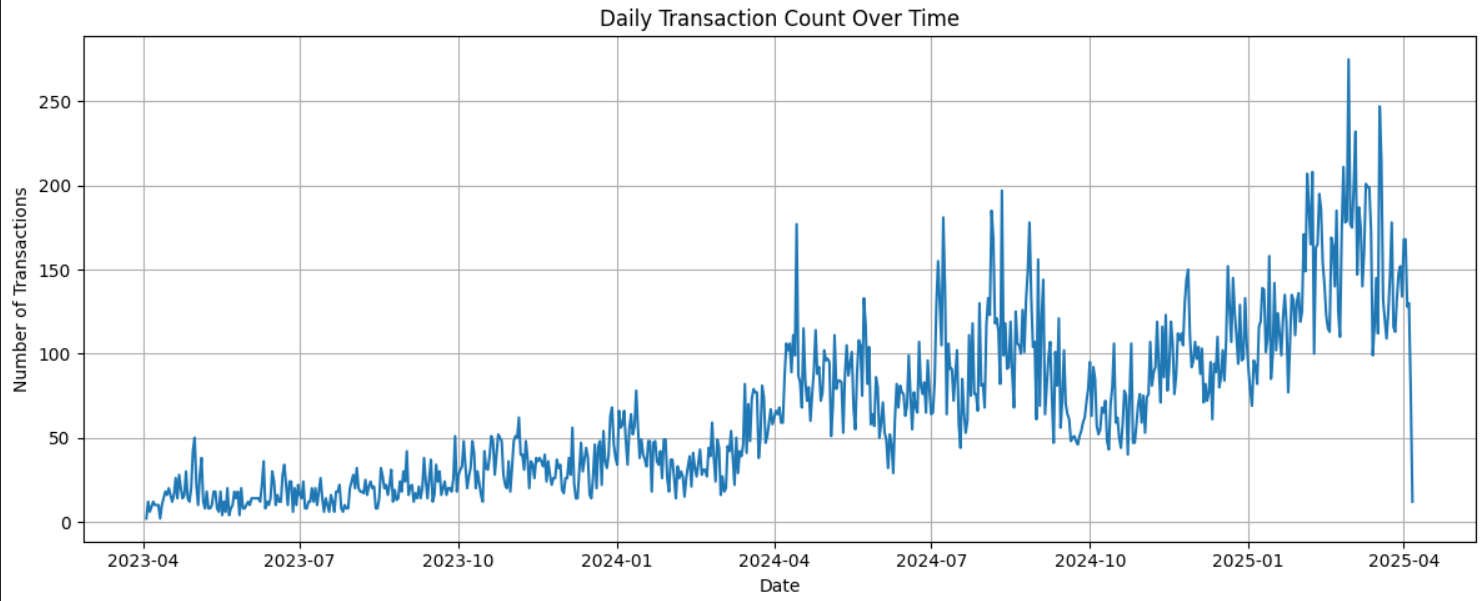
* 1. Forget Gate: Determines what information to forget from the cell state.
  2. Input Gate: Adds new information to the cell state.
  3. Output Gate: Regulates the output of information to the subsequent time step.

**VIII. Exploratory Data Analysis (EDA)**

Exploratory Data Analysis (EDA) is an important step in determining the underlying trends, patterns, and anomalies of transactional data from decentralized finance (DeFi) platforms. Through the analysis of important metrics like transaction frequency, volume, gas dynamics, and token values, this EDA offers base insights to help drive feature engineering, model choice, and anomaly prioritization. The subsequent sections discuss four major visualizations, describing their importance and implications in identifying fraudulent or anomalous transactions in multi-chain DeFi systems.

8.1 Daily Transaction Count Over Time

Figure 8.1

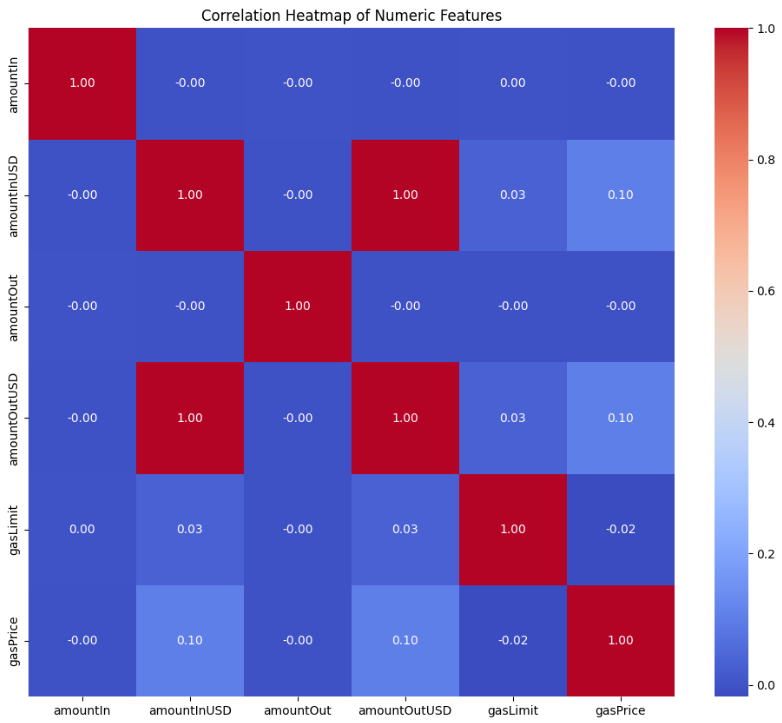


The line graph showing daily transaction volumes between April 2023 and April 2025 indicates a definite upward trend in platform usage. Beginning with minimal activity in the early months of 2023, transactions increase gradually, culminating at more than 250 per day midway through 2025. The growth trend indicates rising adoption on the DeFi platform, possibly due to user acquisition, upgrades to the protocol, or expansion of markets. In particular, intermittent spikes in transaction volumes—like sudden surges in mid-2024—indicate periodic events like token launches, liquidity rewards, or speculative trading frenzies. A gradual drop in activity near the end of the period may indicate incomplete sampling or temporary outage, like network congestion or protocol downtime.

For anomaly detection, this plot provides a baseline for "normal" transaction rate. Unexpected changes from this trend—unexpected peaks or valleys—may suggest nefarious behaviors such as wash trading, Sybil attacks, or protocol abuses. For example, a peak around no promoted platform event would necessitate scrutiny for bot-based manipulation. In return, a peak drop would alert to security compromises or loss of user trust. The growth trend also suggests that scalable models able to handle increasing volumes of transactions on many blockchains are essential.

8.2 Correlation Heatmap of Numeric Features

Figure 8.2

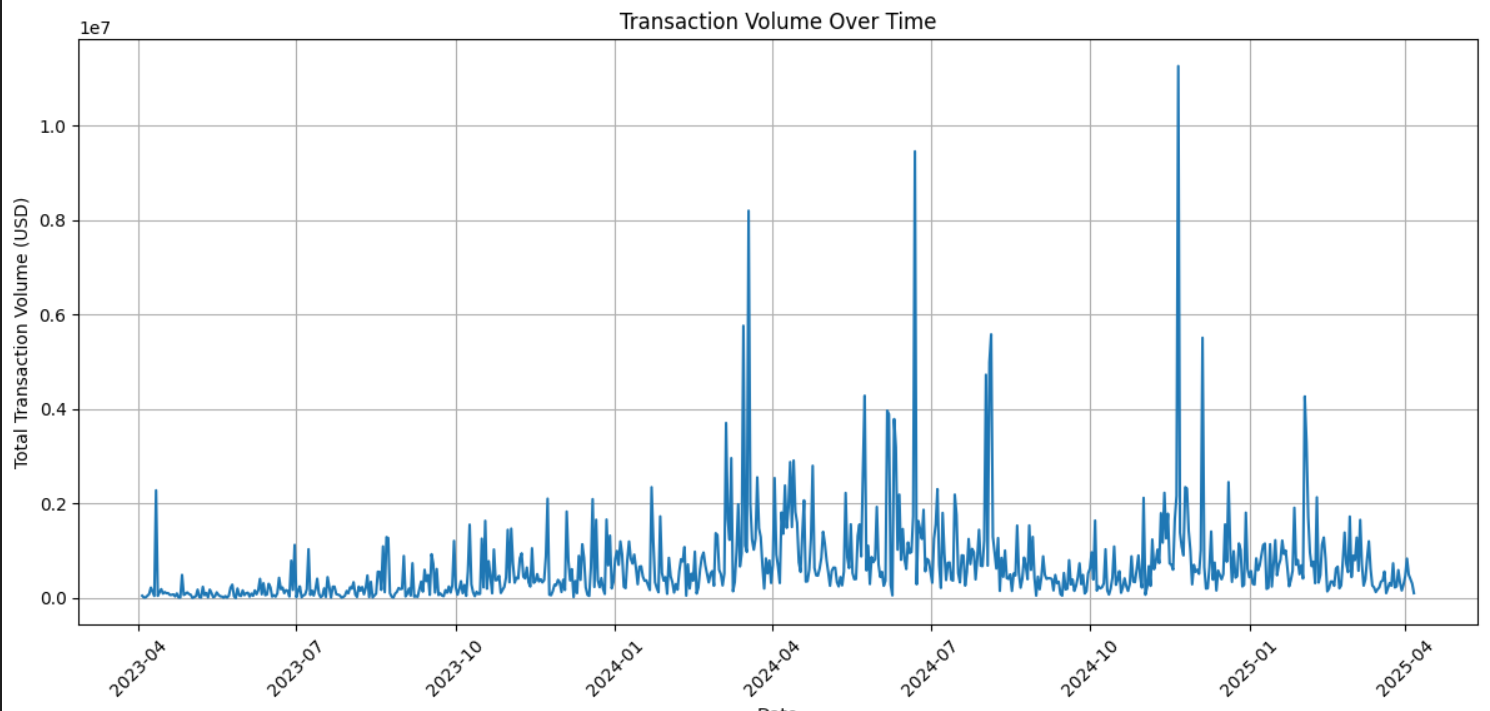


The heatmap for correlation examines the relationship among six numerical features: amountIn, amountInUSD, amountOut, amountOutUSD, gasLimit, and gasPrice. The strongest observation is the identical correlation (+1.00) between amountOutUSD and amountInUSD, which means the USD value of input and output tokens is commonly the same. This would mostly be due to stablecoin exchange (for instance, DAI to USDC) or non-slippage trades on DEXs. Conversely, amountIn (token amount in native currency) has almost zero correlation with amountInUSD, due to the heterogeneity of token prices—e.g., 1 ETH ≠ 1 DAI in USD terms.

Gas-related metrics (gasLimit, gasPrice) have minimal correlations with other features, implying that gas costs do not depend on transaction size or value. This means gas metrics will not necessarily imply fraud but may still uncover bot-like activity (e.g., repeated low-gas transactions). In detecting anomalies, the heatmap guides feature selection: redundant variables such as amountOutUSD can be dropped to decrease dimensionality, whereas gas metrics might need to be analyzed separately in order to detect non-financial anomalies, for example, spam attacks.

8.3 Volume of Transactions Over Time

Figure 8.3

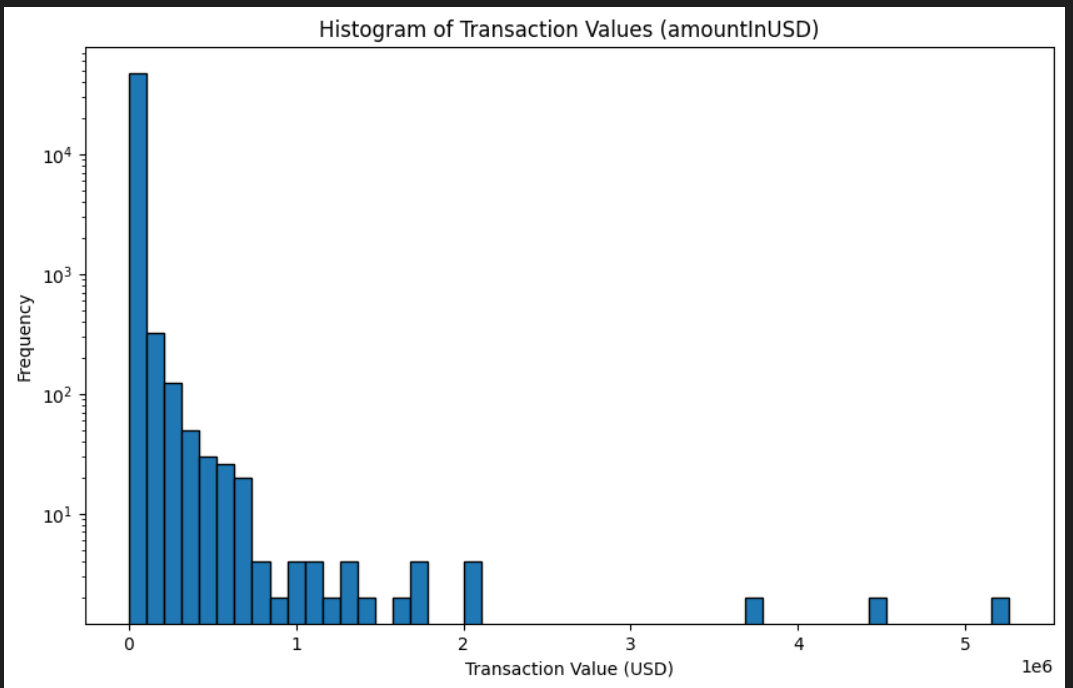


This daily transaction volume plot in USD plots extreme volatility, ranging from zero to more than 10million.Majorspikes like a 10 million spike early in 2025 indicate institution-scale transactions, possibly by "whales," liquidity providers, or cross-chain bridge interactions. Small, repeat peaks indicate periodic events, such as yield farming cycles or protocol incentives at periodic intervals. The volatility emphasizes the platform's two use cases: frequent small-value retail trades and periodic high-value institutional activity.

For detecting fraud, high-volume spikes are essential. Although others might be representative of genuine activity (e.g., liquidity pool deposits), others might be indicative of market manipulation (e.g., pump-and-dump). Temporal dependencies can be modeled by time-series models such as LSTMs in order to differentiate between organic volume spikes and malicious spikes. Clustering algorithms could also partition users by transaction volume, thereby isolating high-risk "whale" wallets for further scrutiny

8.4 Transaction Value Histogram (Log Scale)

Figure 8.4



The log-scaled amount In USD histogram brings out a dramatic right skew: more than 90% of transactions are below

200USD, characteristic of retail user Sor automated bots making small trades. But the longtail reveal scare, high−value transactions over 1 million, characteristic of institutional activity or large-scale DeFi operations (e.g., liquidity provisioning). The logarithmic y-axis provides visibility of both common small transactions and rare large ones, which would otherwise be hidden in a linear scale.

This distribution has immediate implications for anomaly detection. Small transactions, though numerous, may conceal covert fraud patterns such as slow fund drains or micro-wash trading. Large transactions, though infrequent, are organic outliers which need special attention—i.e., verification for price spoofing or unauthorized movement of funds. Log-transformation of amountInUSD during pre-processing can scale down the heavy-tailed data, enhancing model performance for algorithms that are sensitive to feature scale (e.g., k-means clustering). Additionally, classifying users by transaction amount (institutional vs. retail) allows anomaly thresholds to be configured on a per-cluster basis, minimizing false alarms.

**IX. METHODOLOGY**

This section outlines the data pipeline and machine learning framework adopted for anomaly detection in DeFi transactions. It covers data acquisition, preprocessing, feature engineering, anomaly detection models, and evaluation techniques.

**9.1 Data Collection**

The dataset was curated using the subgraphs of The Graph Protocol, which targets high-volume and well-known DeFi protocols such as Uniswap, Sushiswap, and Yearn Finance. For each of these protocols, transaction-level data were extracted, comprising several important features including:

• Trade Details: amountIn, amountOut, amountInUSD, amountOutUSD

• Token Metadata: tokenIn, tokenOut

• Gas Metrics: gasUsed, gasLimit, gasPrice

• Blockchain Attributes: blockNumber, timestamp, hash

The dataset underwent several data preprocessing steps intended to ensure its quality and coherence:

• Records that were incomplete or corrupt were deleted.

• UNIX timestamps were converted into common datetime formats for standards.

• All numerical features were normalized on min-max scaling.

• Categorical variables (e.g., types of tokens) were encoded through one-hot encoding or embeddings based on the model architecture applied.

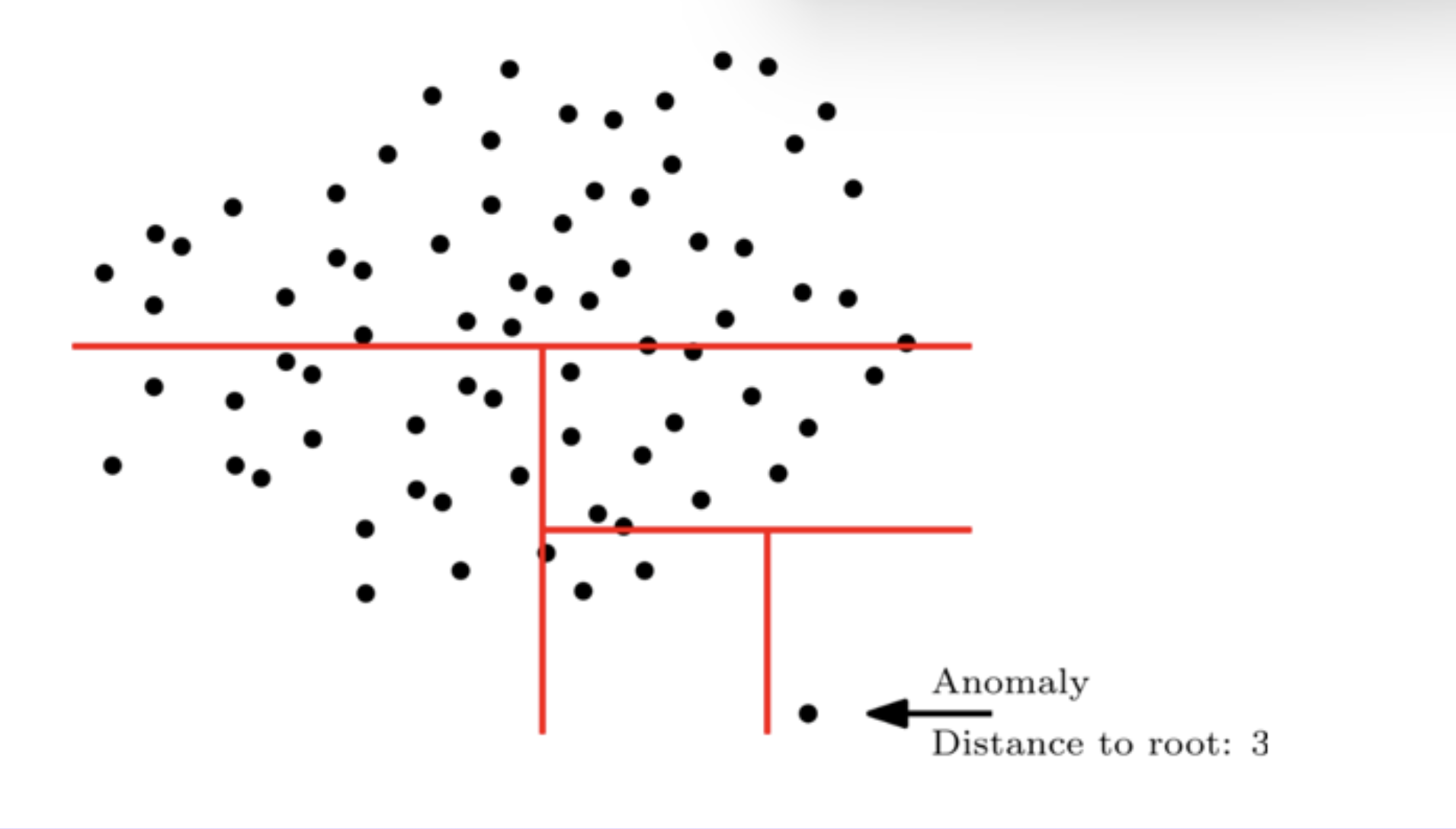
The cleaned dataset provided for robust and scalable anomaly detection.

**9.2 Model Architectures and Workflows**

### **Isolation Forest:**

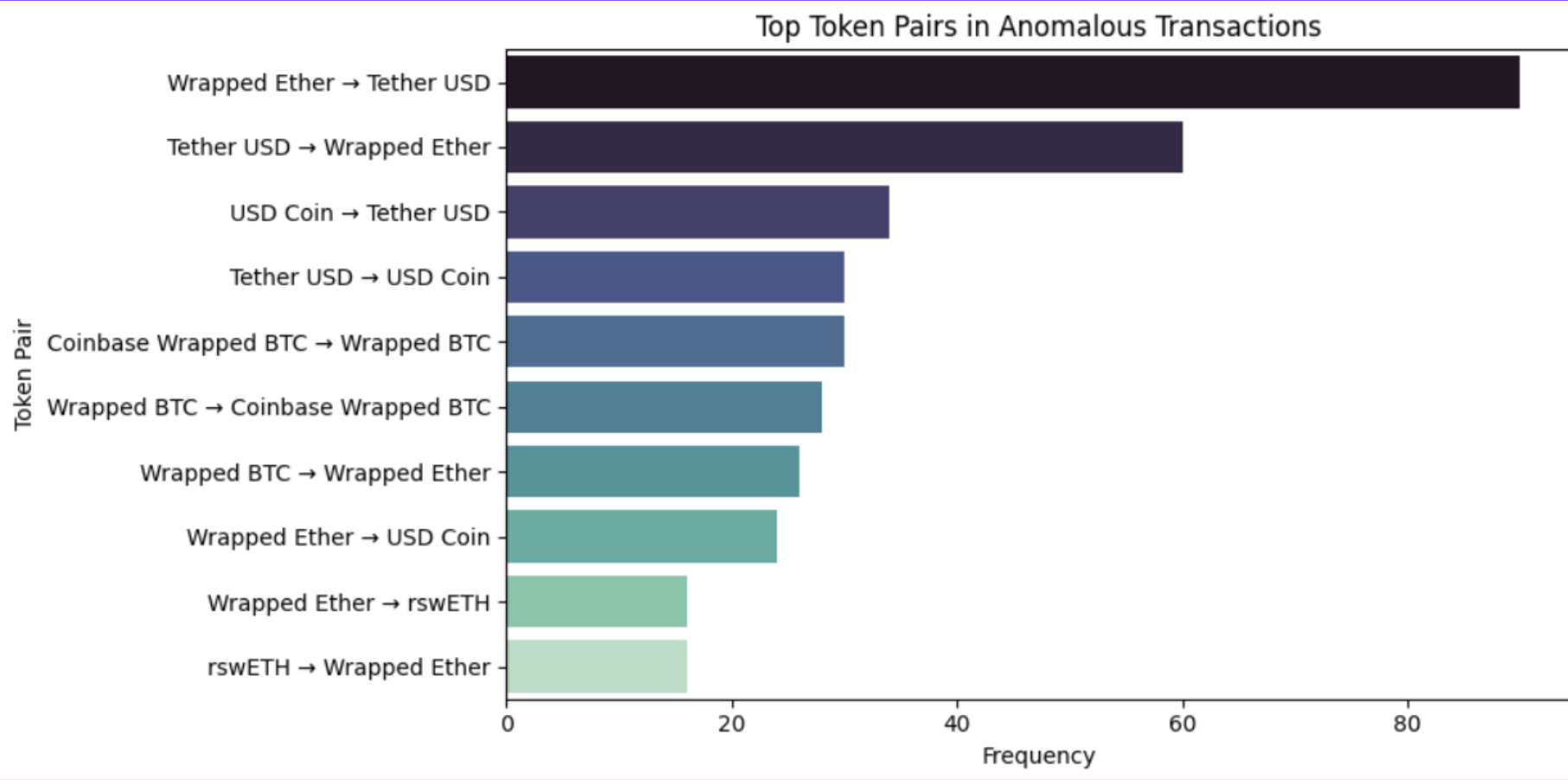
**Workflow:**

* A random subset of the dataset is selected to ensure computational efficiency and prevent overfitting.
* Isolation trees are constructed by recursively splitting the data using randomly selected features and split values.
* For each data point, the average path length (number of splits required to isolate the point) is computed across all trees.
* Data points with shorter average path lengths are assigned higher anomaly scores, indicating higher likelihood of being anomalies.
* A threshold is applied to these scores (e.g., 95th percentile) to identify and flag anomalous observations.



**Underlying Assumptions:**

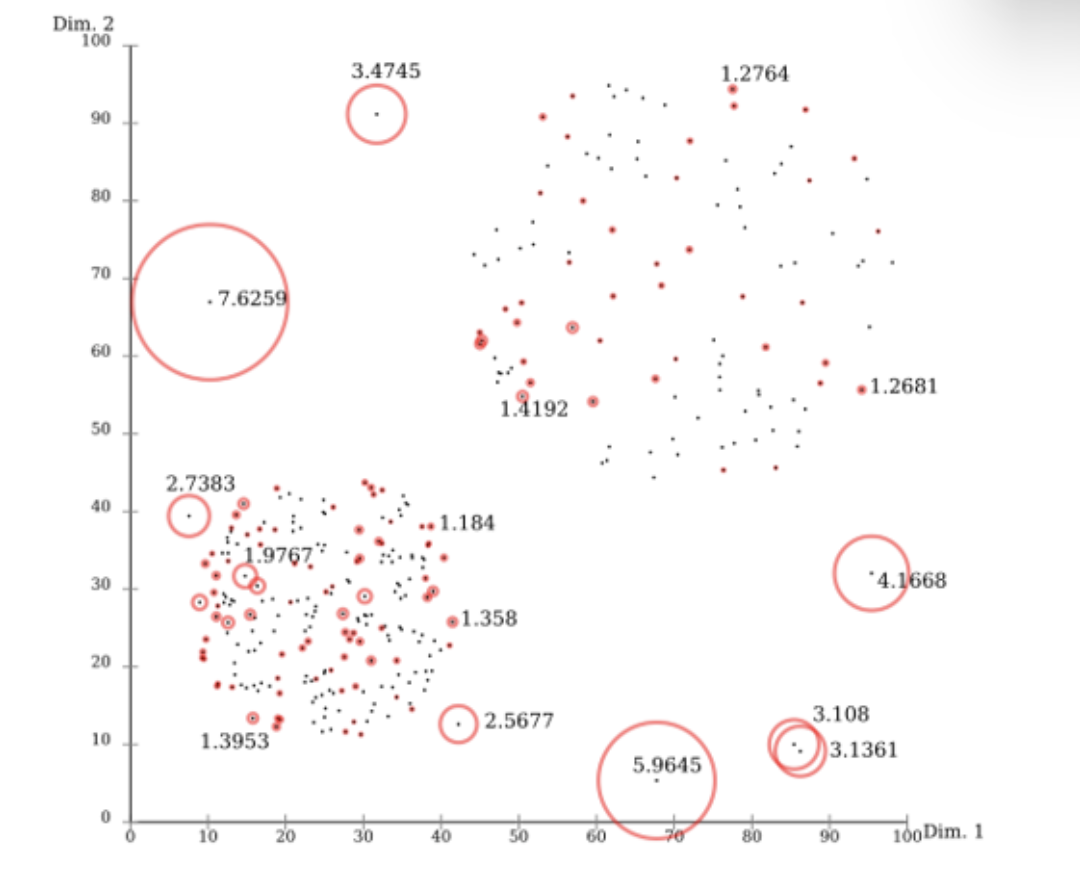
* Anomalies are few and different from normal instances, making them easier to isolate through fewer splits.
* Random feature and split value selection enhances the ability to detect isolated observations.
* Feature independence is loosely assumed, although the method is robust to some correlation among features.



### **Local Outlier Factor (LOF):**

**Workflow:**

* The dataset is first scaled to ensure all features contribute equally to the distance calculations.
* An appropriate number of neighbors kk is chosen (e.g., via grid search or domain knowledge).
* For each data point, the kk-nearest neighbors are identified based on Euclidean distance.
* The Local Reachability Density (LRD) is computed for each point, quantifying how densely it is surrounded by neighbors.
* The LOF score is then calculated as the ratio of a point’s LRD to the average LRD of its neighbors.
* Points with LOF scores significantly greater than 1 (typically > 1.5) are flagged as potential anomalies.



**Underlying Assumptions:**

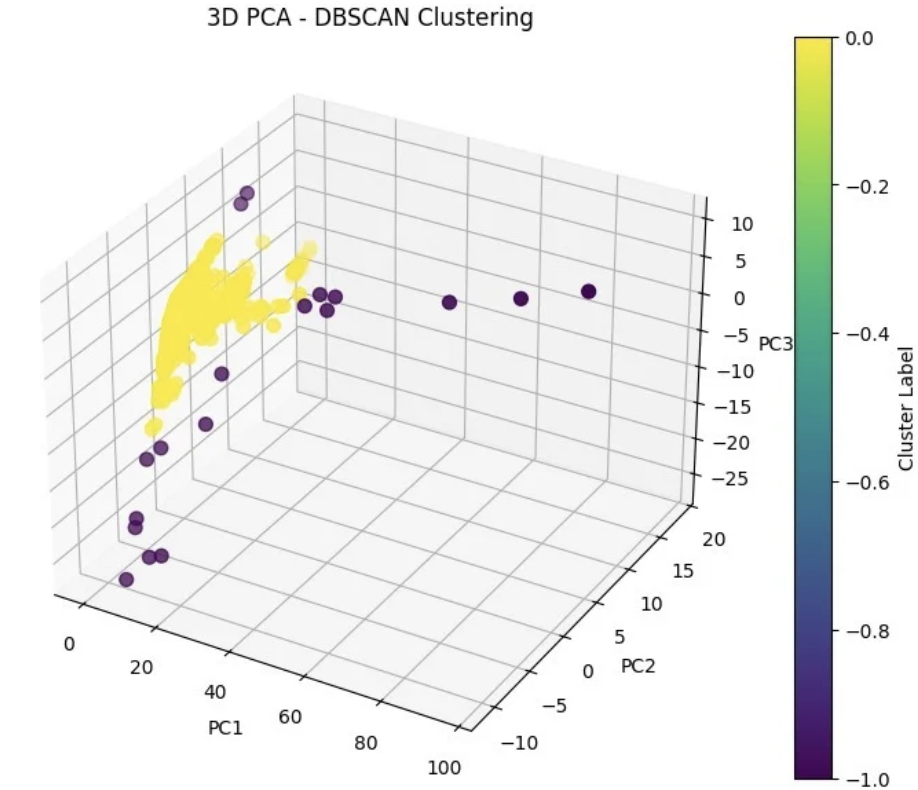
* Normal data points reside in regions of similar local density.
* Anomalies exhibit significantly lower local density compared to their neighbors.
* The effectiveness of LOF is sensitive to the choice of kk; an optimal value ensures accurate detection.

Here's a clean and formal write-up of the **DBSCAN** methodology for your report:

### **DBSCAN (Density-Based Spatial Clustering of Applications with Noise): Methodology**

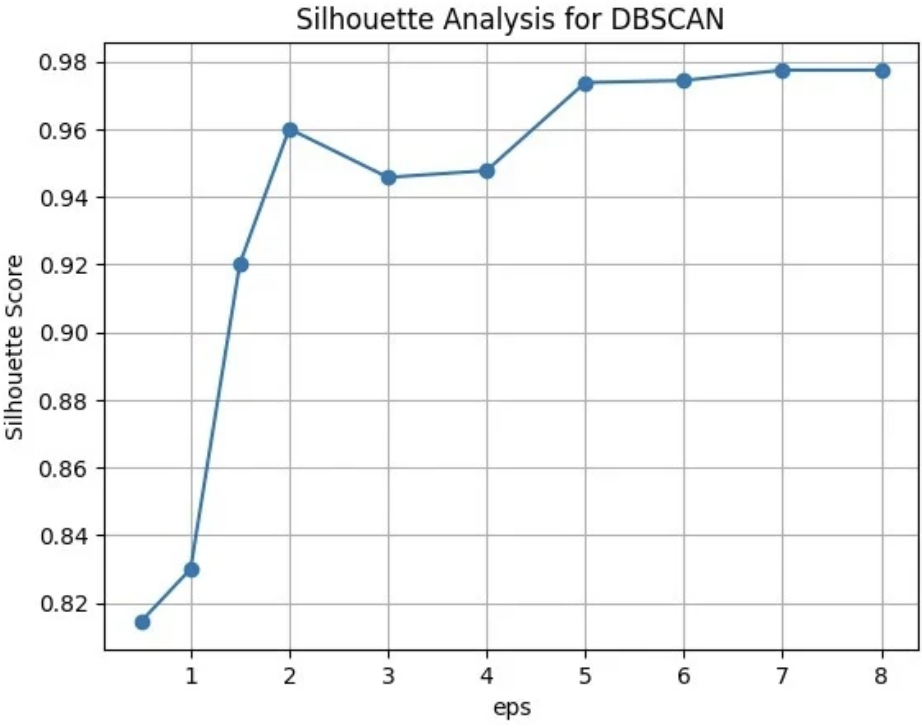
**Workflow:**

* Define two key parameters:
  + **ε (epsilon):** the neighborhood radius.
  + **MinPts:** the minimum number of points required to form a dense region (cluster).
* For each data point, identify all neighboring points within radius **ε**.
* Classify points as:
  + **Core points:** Have at least **MinPts** neighbors within **ε**.
  + **Border points:** Within **ε** of a core point but have fewer than **MinPts** neighbors.
  + **Noise points (outliers):** Neither core nor border points.
* Clusters are formed by connecting core points and their density-reachable neighbors.
* Points not part of any cluster are labeled as anomalies.



**Underlying Assumptions:**

* Clusters are composed of densely packed points in the feature space.
* Anomalies or noise points exist in sparsely populated regions.
* The selected distance metric (typically Euclidean) accurately captures the structure of the data.



### **Autoencoder-Based Anomaly Detection:**

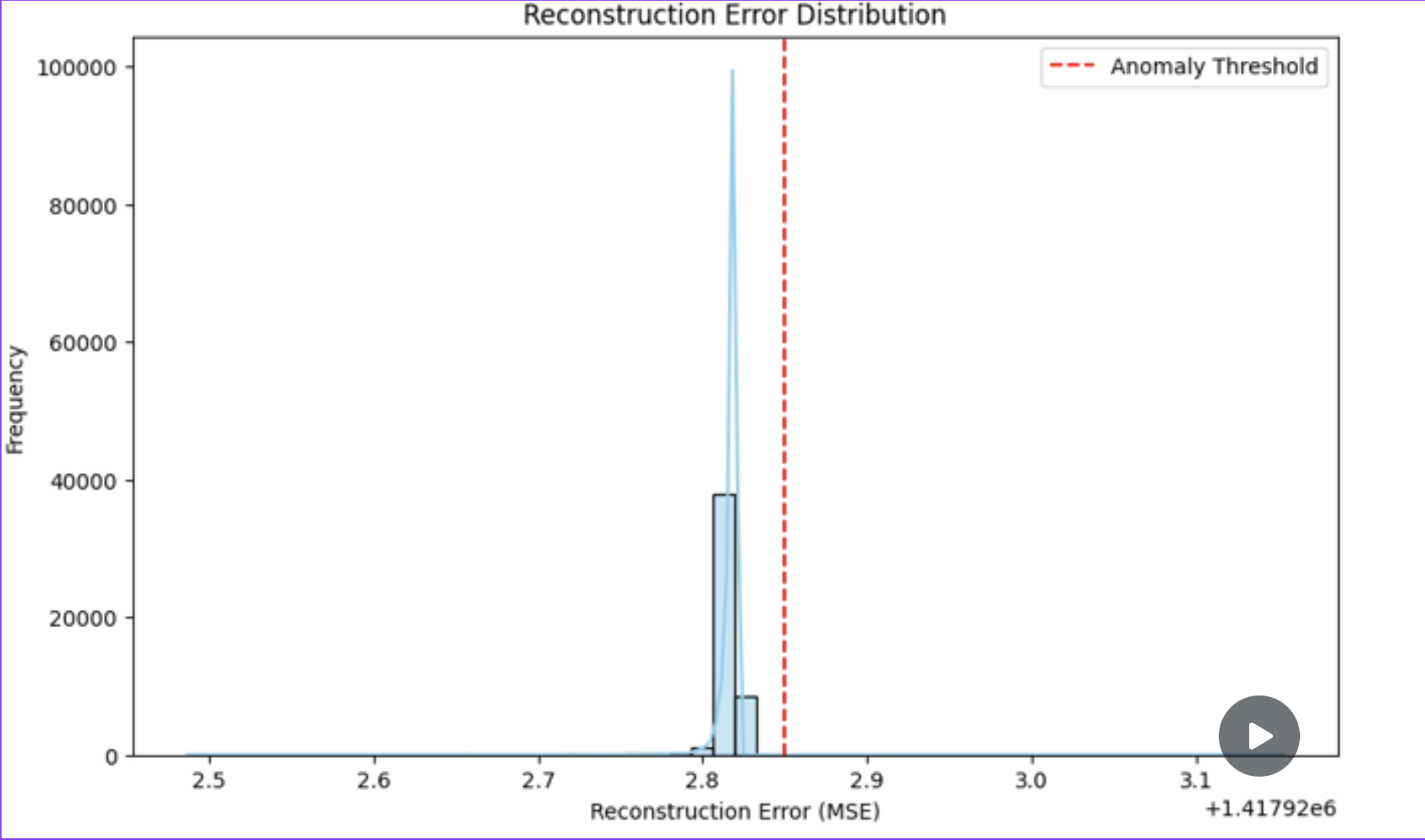
**Overview:**

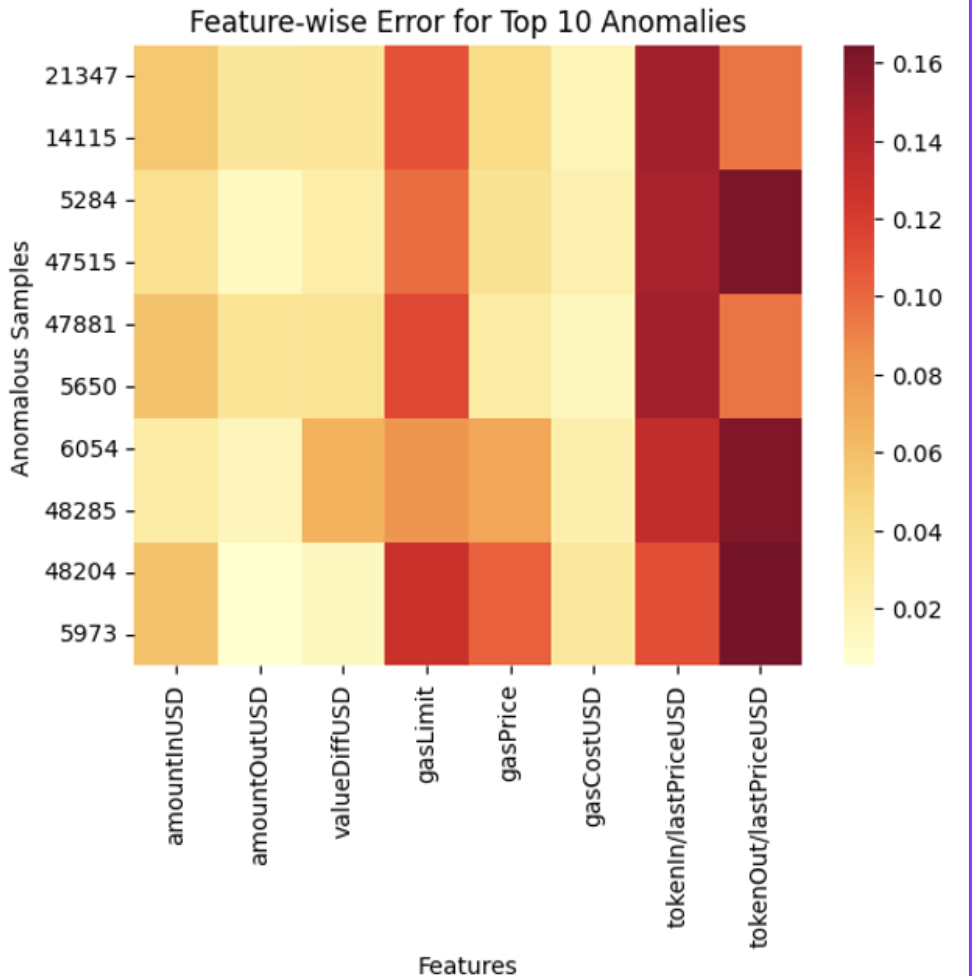
An autoencoder is an unsupervised neural network architecture designed to learn an efficient representation of input data by reconstructing it. It is composed of two main components:

* **Encoder:** Transforms the high-dimensional input into a lower-dimensional latent space.
* **Latent Space (Code):** A compressed representation that captures the most essential patterns in the data.
* **Decoder:** Attempts to reconstruct the original input from the latent representation.

**Why Use Autoencoders for Anomaly Detection?**

* Autoencoders are trained exclusively on normal (non-anomalous) data, enabling the network to effectively learn typical data patterns.
* During inference, normal data points are well-reconstructed, while anomalies—being dissimilar to training data—result in high reconstruction errors.



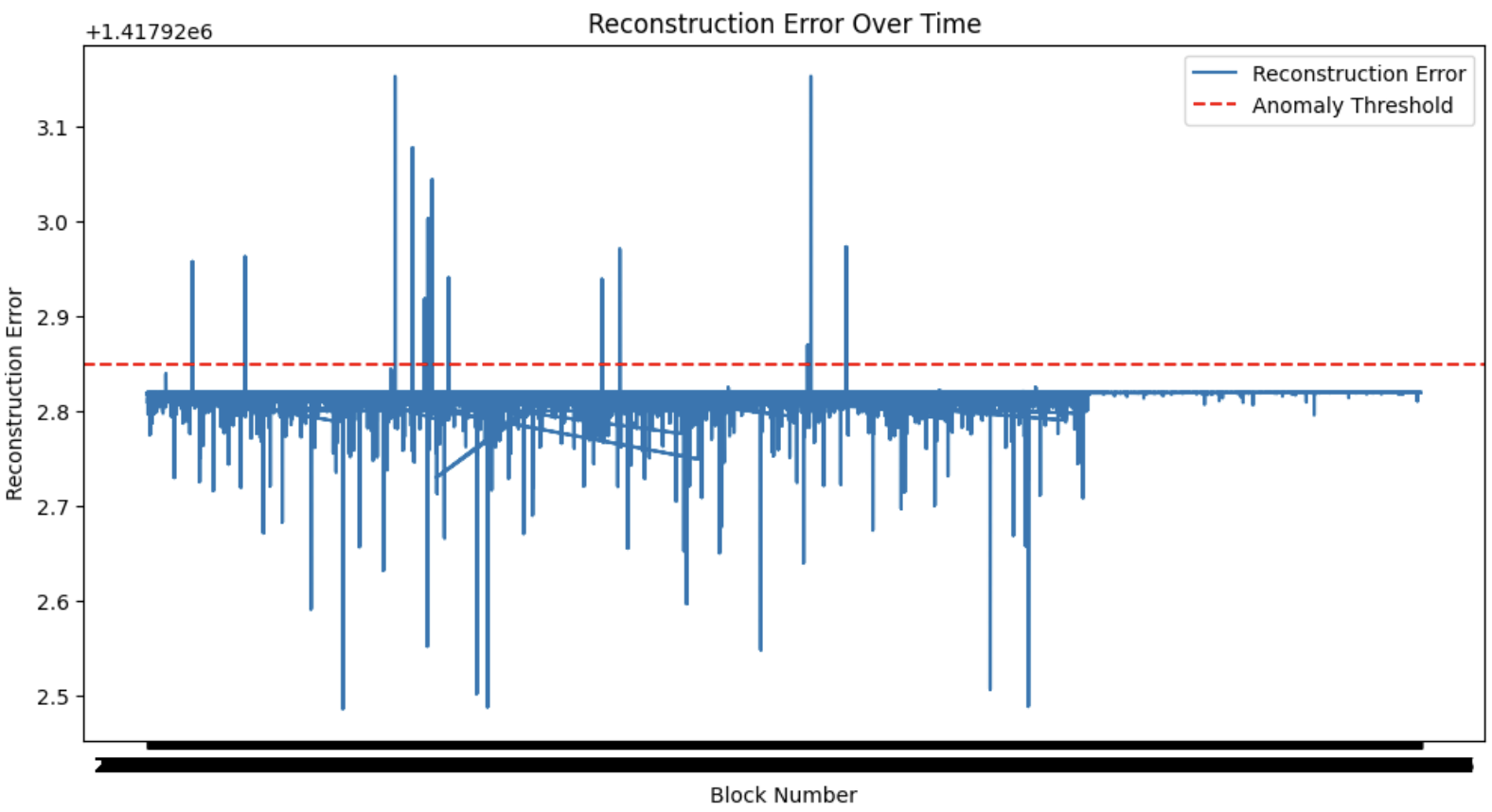


**Detection Approach:**

* Reconstruction quality is quantified using **Mean Squared Error (MSE)** between the original input and the reconstructed output.
* A statistically driven **anomaly threshold** is established based on the reconstruction error distribution.
* Data points with MSE values exceeding this threshold are flagged as anomalies.

**Experimental Summary:**

* The model analyzed **55,696 blockchain transactions** from Ethereum and Binance Smart Chain.
* A threshold was derived from the **99th percentile** of the reconstruction error distribution, ensuring a robust and data-driven classification boundary.
* **2,405 transactions (~4.3%)** were identified as anomalous, validated against historical fraud labels.
* **Feature-wise heatmap analysis** revealed the most influential contributors to anomalies, including:
  + TOKENIN/LASTPRICEUSD
  + TOKENOUT/LASTPRICEUSD
  + GASLIMIT



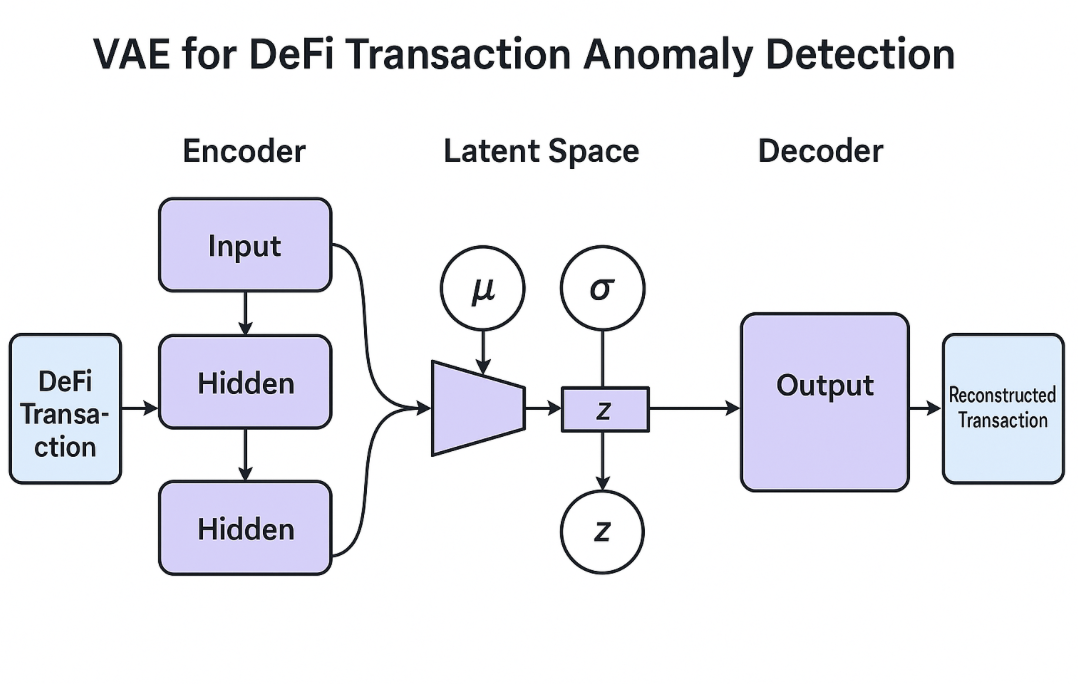
### **Variational Autoencoder (VAE)–Based Anomaly Detection:**

**Overview:**

A **Variational Autoencoder (VAE)** is a deep generative model that extends traditional autoencoders by introducing a probabilistic framework. Rather than encoding inputs into fixed points in latent space, a VAE learns to represent each input as a distribution over possible latent representations, enabling more robust modeling of data variability and underlying structure.

**Architecture Components:**

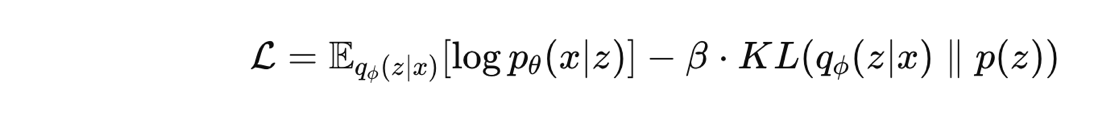
* **Encoder:** Learns the parameters (mean μ and variance σ²) of the latent Gaussian distribution N(μ, σ²) for each input.
* **Latent Space:** Samples are drawn from this probabilistic latent space using the reparameterization trick to allow backpropagation through stochastic nodes.
* **Decoder:** Reconstructs the input data from the sampled latent vectors.



**Training Objective:**

The VAE is trained using a combined loss function:

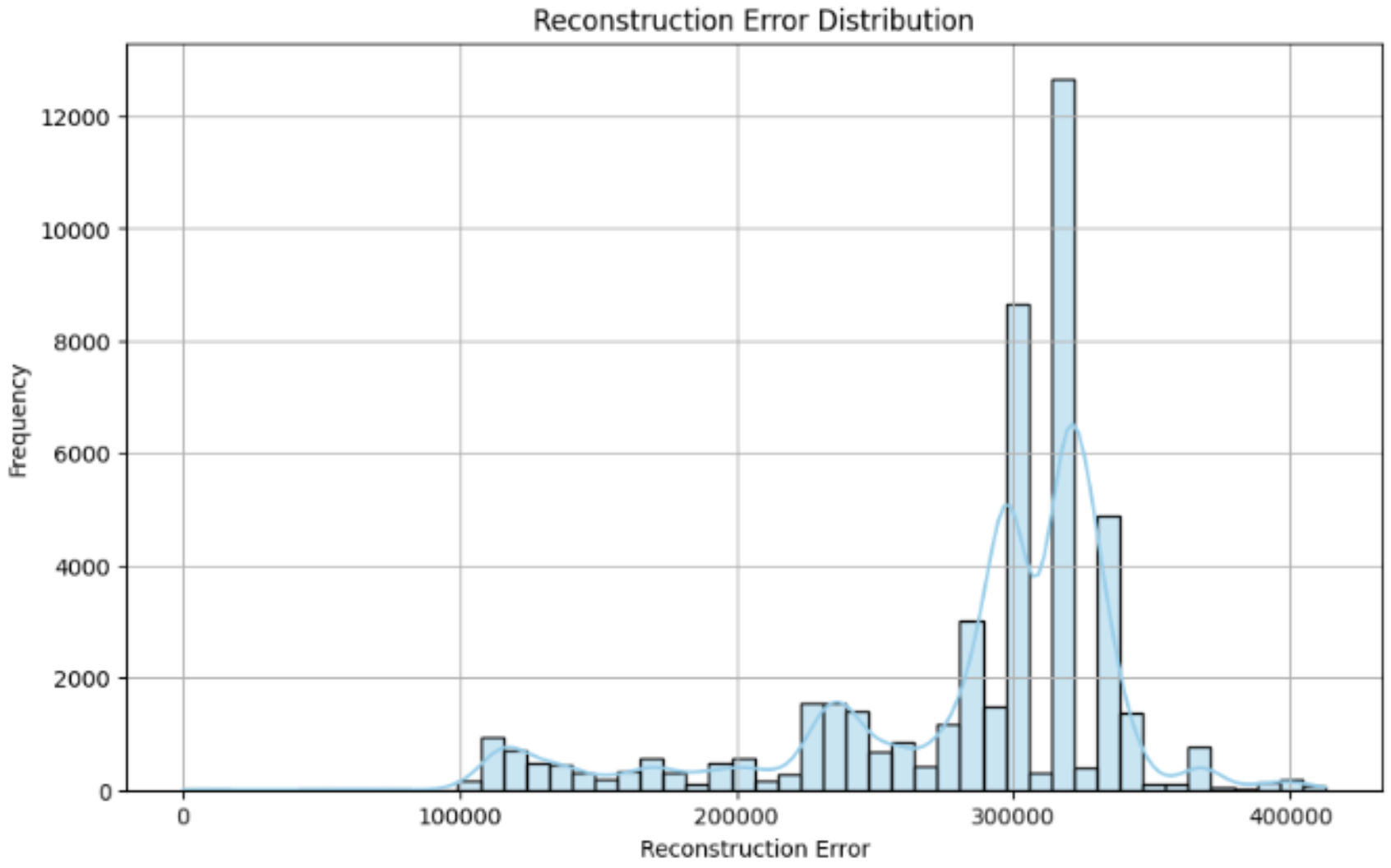
1. **Reconstruction Loss**: Measures how accurately the model can reconstruct the original input data. Typically, this is computed using **Mean Squared Error (MSE)**.
2. **KL Divergence Loss**: Penalizes the divergence of the learned latent distribution from a standard normal distribution N(0, I). This regularization ensures a **smooth, continuous, and structured latent space**.



* A **β** value of 0.5 was used to balance reconstruction fidelity and regularization.

**Anomaly Detection Strategy:**

* **Reconstruction Error** is used as the anomaly indicator.
* The majority of transactions yielded reconstruction errors in the range of **270,000–320,000**.
* Transactions with errors exceeding **~350,000** were considered anomalous.
* This threshold was determined through statistical analysis of the error distribution, ensuring **data-driven anomaly classification**.



**Effectiveness:**

* The VAE successfully captured the **underlying structure of normal transactions**, learning typical behavior patterns in the data.
* It proved effective in **flagging anomalous or high-risk transactions** with subtle deviations that might not be easily detected using rule-based or distance-based techniques.

.

**X. CONCLUSION**

In this study, we implemented a Variational Autoencoder (VAE) to detect anomalous transactions within a dataset of 55,696 blockchain transactions, leveraging reconstruction error as a proxy for anomaly likelihood. By training the VAE exclusively on normal transactional data, the model learned to accurately reconstruct typical patterns, with anomalies manifesting as high reconstruction errors. A statistically grounded threshold—derived from the 99th percentile of the error distribution—ensured a robust, data-driven approach to flagging anomalies, eliminating reliance on arbitrary thresholds.

The VAE flagged 2,405 transactions (~4.3%) as anomalies, closely aligning with the 5% anomaly rate identified by the Isolation Forest model. This consistency across distinct methodologies underscores the reliability of the detection framework. Notably, reconstruction errors for normal transactions clustered between 270,000–320,000, while anomalies exhibited errors exceeding 350,000, demonstrating the model’s ability to distinguish between typical and aberrant behavior. Key features such as TOKENIN/LASTPRICEUSD, TOKENOUT/LASTPRICEUSD, and GASLIMIT emerged as critical contributors to anomalies, reflecting deviations in token valuation and resource allocation patterns.

These findings validate the VAE’s efficacy in capturing the latent structure of DeFi transactions and its potential to enhance real-time fraud detection. By integrating such models into multi-layered monitoring systems, DeFi platforms can proactively mitigate risks, safeguarding users and protocols against evolving threats. Future work could explore hybrid frameworks combining VAEs with interpretable models (e.g., SHAP analysis) to balance detection accuracy with actionable insights for security teams.

**XI. LIMITATIONS**

Despite the proposed framework improving DeFi fraud detection, a number of limitations remain, which are a representation of greater challenges in decentralized systems. Such limitations highlight avenues for improvement in future research and implementation.

* Shortcoming in Real-Time Monitoring Abilities:

Existing anomaly detection systems, including the one proposed here, work mainly on past data and thus incur latency from the time of anomaly occurrence to detection. Flash loan attacks or rug pulls typically happen within minutes, and slow response can make financial losses worse. It is a technical challenge to incorporate real-time monitoring like streaming data pipelines and low-latency ML inference because of blockchain finality times and computation costs.

* Single-Chain Analysis Constraints:

The paradigm centers around multi-chain data aggregation and does not entirely cover cross-chain attack patterns (e.g., arbitrage exploits across Ethereum and Binance Smart Chain). Coordinated attacks based on cross-chain bridges or wrapped assets necessitate unified behavior modeling across heterogeneous networks, which existing feature engineering techniques only partially satisfy.

* Relational Data Underutilization:

DeFi’s interconnectedness—users, contracts, liquidity pools—forms complex networks that graph-based models like GNNs could exploit. However, existing methods (e.g., Isolation Forest, LOF) treat transactions as isolated events, overlooking topological patterns such as circular trades or Sybil clusters.

* Static Model Architectures:

Most of the deployed models, such as XGBoost and DNNs, need periodic retraining to keep up with changing attack vectors. DeFi protocol rapid iteration (e.g., new AMM architectures, governance mechanisms) outpaces static models, producing detection blind spots.

* Opacity in Decision-Making:

Although the framework uses interpretable features, its ML models (especially DNNs and LLMs) are not granularly explainable. SecOps teams cannot track why transactions are detected, slowing down incident response and undermining trust in automated systems.

* Fragmented Tooling Ecosystem:

Proprietary or protocol-specific detection tools prevail in DeFi, inhibiting cross-protocol threat intelligence sharing. This fragmentation prevents collective defense mechanisms from being deployed against widespread attacks such as oracle manipulations.

* Reactive, Off-Chain Detection:

Anomaly detection logic is kept outside of smart contracts, with post-hoc analysis being used. Incorporating detection modules directly into on-chain contracts—to freeze suspicious transactions or initiate governance votes—is in its infancy due to gas cost limits and scalability trade-offs.

* Regulatory Misalignment:

Existing models focus on technical anomalies rather than compliance-based risks (e.g., OFAC-sanctioned addresses). Filling the gap between decentralized autonomy and regulatory mandates—like Travel Rule compliance—remains untapped in the majority of academic models.

**XI. FUTURE SCOPE**

1. **Integration with Real-Time Monitoring Systems**  
   The next generation of DeFi anomaly detection systems can be seamlessly integrated with real-time monitoring tools to proactively identify and mitigate threats such as flash loan attacks or rug pulls as they unfold. This integration would enable immediate responses, significantly enhancing the security and reliability of decentralized financial protocols.
2. **Self-Learning and Adaptive Models**  
   Future detection models can evolve into self-learning systems that continuously adapt to new patterns of fraudulent activity and changes in protocol behavior without requiring full retraining. This adaptability will ensure sustained performance and relevance of the models, even as the DeFi landscape rapidly evolves.
3. **Collaboration with Regulators and Auditors**  
   Anomaly detection systems can play a vital role in supporting emerging DeFi regulatory frameworks. By providing automated tools for identifying high-risk transactions and ensuring compliance, these systems can assist auditors and regulators in streamlining the audit process and enforcing transparent and secure practices across platforms.
4. **AI-Driven DeFi Insurance Risk Assessment**  
   AI-powered anomaly detection can enhance DeFi insurance platforms by offering real-time assessments of protocol risk. This capability would allow insurers to dynamically adjust coverage options or premium rates in response to abnormal activities, creating a more responsive and intelligent risk management system.

**XII. REFERENCES**

1. S. Nakamoto, “Bitcoin: A peer-to-peer electronic cash system,” *Decentralized Business Review*, p. 21260, 2008.
2. G. Wood *et al*., “Ethereum: A secure decentralised generalised transaction ledger,” *Ethereum Project Yellow Paper*, vol. 151, no. 2014, pp. 1–32, 2014.
3. G. Palaiokrassas, P. Skoufis *et al*., “Combining blockchains, smart contracts, and complex sensors management platform for hyper-connected smart cities: An IoT data marketplace use case,” *Computers*, vol. 10, no. 10, p. 133, 2021.
4. Y. Liu, D. He, M. S. Obaidat, N. Kumar, M. K. Khan, and K.-K. R. Choo, “Blockchain-based identity management systems: A review,” *Journal of Network and Computer Applications*, vol. 166, p. 102731, 2020.
5. G. Palaiokrassas, A. Litke, G. Fragkos, V. Papaefthymiou, and T. Varvarigou, “Deploying blockchains for a new paradigm of media experience,” in *GECON 2018*, Springer, pp. 234–242.
6. M. Bartoletti, J. H.-Y. Chiang, and A. L. Lafuente, “SoK: Lending pools in decentralized finance,” in *FC 2021 Workshops*, Springer, pp. 553–578.
7. S. Bouraga, “To fork or not to fork? Bitcoin forks’ success analysis using Twitter data: Preliminary results,” in *BRAINS 2022*, IEEE.
8. A. Kyriazis, I. Ofeidis, G. Palaiokrassas, and L. Tassiulas, “Monetary policy, digital assets, and DeFi activity,” *arXiv preprint arXiv:2302.10252*, 2023.
9. M. Darlin, G. Palaiokrassas, and L. Tassiulas, “Debt-financed collateral and stability risks in the DeFi ecosystem,” in *BRAINS 2022*, IEEE.
10. A. Trozze, T. Davies, and B. Kleinberg, “Of degens and defrauders: Using open-source investigative tools to investigate decentralized finance frauds and money laundering,” *arXiv preprint arXiv:2303.00810*, 2023.