Decentralized Finance Final Project Report

Proposed Model For DeFi Accountability

Kaleab Belete (kaleab@berkeley.edu)[1], Rohullah Najibi (Rohullahnajibi@berkeley.edu)[1], Tilek Chubakov (tchubakov@berkeley.edu)[1], Shiv Sethi (shiv_sethi@berkeley.edu)[1], Yue Hu (yuehu@berkeley.edu)[1], and Amal Thiru (adthiru@berkeley.edu)[1]

[1]University of California, Berkeley

December 12, 2022

1 Abstract

One of the biggest hurdles decentralized finance faces is the lack of accountability which makes investors and users hesitant to jump into the space. A key contributor is the large amounts of scams in the space and a general lack of trust that dissuades people from using crypto and DeFi tech.

We aim to improve trust in decentralized finance, by building a set of tools that can score DeFi items for legitimacy while providing some sort of baseline analytics (Score, Confidence, etc). There are active analysis-type projects out already but they are mostly built for the use of law enforcement to manually analyze transitions and there are some ongoing attempts to make some sort of legitimacy ranting but nothing has stuck. Our aim is to provide a unified tool for basic DeFi accountability that gives users a baseline idea of the risks involved in their actions. This paper focuses on research conducted on these scams and potential scam-tracking solutions to make the DeFi space a more fair and trustworthy environment for everyone.

2 Introduction

Since the inception of the ideas of traditional centralized finance, or CeFi, began centuries ago, the world has been struggling to deal with scams, fraud, and trust with its assets. For most, this meant that a single fire, drought, or theft in their homes would eradicate their entire life's savings and force them to start again. Fortunately, with the emergence of technology and traditional banks, many of these risks were averted. People's money was safeguarded and their assets were insured. However, as new solutions arose, so did new means of thievery and scamming. And thus, by means of identity fraud or other lies to trap innocent people, scammers get people to transfer over portions of their wealth. Sophisticated scammers who know their trade well are even able to do this with little risk of being caught. This, coupled with centralized banks themselves having flaws led to the emergence of Decentralized Finance – DeFi. As the DeFi space grew, so did the opportunities in the space allowing many to have freedom over their wallets which they never did in a CeFi system. However, as the DeFi space grew, so too did the number and danger of scams in the space. Sophisticated scammers can find ways to use this freedom of anonymity as a tool to institute large scams in manners where they do not get caught.

3 Related Works

There are currently no widely adopted standards for accountability in the DeFi space, making it an excellent frontier for exploration. There are a lot of interconnected components that go into having a system like this work, like digital identity and reputation systems, that have been actively worked on. In the DeFi space, we can also find a lot of parallels to traditional financial systems but there are key differences we must consider.

DeFi gives us unprecedented levels of control over our finances, but it can also come with its own set of risks and drawbacks. When working with DeFi issues of identity, membership, and legitimacy are amplified. Concerns of who is trustworthy and who is a legitimate member of an online community are important

for everything ranging from governance voting to the amount of collateral needed to take out a loan. A Lot of important concepts are involved in the creation of a valid DeFi ID and building up the tooling needed for accountability, we have provided references to the papers and projects that go into these topics below.

Digital Proof of Personhood (PoP) [1] methods aim to improve accountability with a minimal negative impact on security, privacy, or confidentiality. The proposed pseudonym party PoP aims to leverage the security of attendance at an in-person event with the anonymity of a cryptographic digital PoP token. This involves frequent meetings and has a lot of overhead but it seems to work best as a solution for community governance. The author acknowledges there are issues but should slow down the rate of scams by building trust within a community and it would work well on top of the more general system we are proposing. There are a lot of other PoP methods (also mentioned in the reference) like biometric authentication, government identification, and more that have their own trade-offs and could be included in our more general proposal.

DID (Decentralized Identifier) [2] is a cryptographically secure and machine-verifiable string of values that uniquely identifies a person. DIDs use public and private key pairs to protect information and process permissions and are designed to work across different blockchains. Also, DIDs provide a way for individuals and other device-independent entities to communicate compared to IP addresses, which provide a way for devices to interact. CanDID(referenced below) [3] aims to leverage real-world data from some authority to build a DID with legacy compatibility. The paper goes into detail about the possible implementation but again there is a lot of overhead and it requires compliance from legacy systems. Like with the other identification methods it would do well to cut back on scams but doesn't have the more general applicability we are looking for. Some people(young people) don't have the real-world financial history to allow them to build legitimacy in the DeFi space so this ID needs complementary tooling.

An interesting system is a web of trust (referenced below) [4] which is a cryptographic trust model that can essentially work as a co-signing system that allows people to gain trust from a trusted individual co-signing them. This can also be used as a complementary method to our general proposed system that gives added legitimacy to someone by leveraging the legitimacy of another. As stated before there are no widely adopted protocols to detect legitimacy. Our implementation aims to consolidate our key findings and build a general tool that can leverage additional protocols (PoP, DID, Web-of-Trust, blockchain analysis, traditional finance metrics, etc. . .).

From our research five key performance indicators [5] in DeFi include 1) Total Value Locked (TVL): It is the total number of tokens in a protocol. The less the TVL, the more incredible the project. 2) Token Supply on Exchanges: checking the supply of tokens in centralized exchanges. 3) Token Balance Trends/Movements: One needs to take a look at the balance movements of the token since the token supply doesn't always indicate a large number of withdrawals. 4) Inflation Rate: When looking at the protocol, one needs to make sure that the token is not influenced by inflation and devaluation. 5) The Growth of Unique Addresses: This is a good sign since it shows the growth of the token, however, one user can create multiple accounts, which makes it complicated. The article on scoring protocol [6] risk takes strides in evaluating the risk of a certain avenue in DeFi. It uses a variety of metrics including those mentioned above and even comes to construct a weighted formula for risk assessment. This is incredibly useful in that it allows for a path to assess risk while also verifying the importance of risk scores and assessment strategies.

4 Approach

Our project is a multi-model approach that combines multiple approaches. One approach we explored is we are looking into implementing machine learning as our core assessment mechanism transaction analysis. The other approaches are defining our own assessment mechanism that can directly evaluate different items on the blockchain and evaluate them based on the standards we define and a graph model for anomaly detection. All culminating in a novel and complementary system that gives a robust scoring solution that combines the strengths of each approach. All of our code for the aforementioned methods is in our public GitHub repository[1].
Having various systems at play helps alleviate some of the shortcomings of the current work

[1] https://github.com/BeKaleab/DeFi-ID

out there.

4.1 Databases

We collect adequate labeled data, set up and test our neural network, and build a pipeline that would allow us to update the system with new data as more transactions are processed. We pull labeled data sets from sites like Kaggle (both traditional financial data and blockchain), augmenting data through reformatting and combining datasets, and adding our own data. The other core work is pulling data from the chain, tracking pulled data, and building our metrics, and testing our evaluation metrics.

4.2 ML for fraud detection

We use several supervised ML learning models to predict that an address is fraudulent. We analyze the available transaction and ERC20 token transfer data to come up with model features. The features are based on transaction and token transfer temporal, structural, and value statistics. We then conduct manual data analysis and feature engineering. This results in a simple and fast approach to address classification. These ML models are very fast to train and do inference, and can easily be augmented with new models. Additionally, we are able to use ensembling techniques to combine predictions from several models. It is also extendable to include new features and use new models in the future.

We used two datasets to make sure our model is not overfitting. The first dataset [7] has 9842 wallet addresses while the second dataset [8] has 12147 wallet addresses. Both datasets have a feature called flag indicating whether wallet address is fraud or not. The first dataset has 77.86% non fraud and 22.14% fraud wallet addresses. The second dataset is a combination of 57.60% non fraud and 42.40% fraud wallet addresses. Both datasets have many features, however, we dropped the features with variance zero. (Figure 1) below shows the features highly correlated and less correlated to the main feature flag. We can see that among the remaining features, time difference between first and last transactions, average minimum between received transactions, sent transaction, received transaction, average value sent, ERC20 unique sent address and ERC20 unique received token name are features that are less correlated to the flag while average minimum between sent transaction, number of created contracts, maximum value received, average value received, minimum value sent to contract, total ether sent, total ether balance, ERC20 total ether received, ERC20 total ether sent and ERC20 total ether sent contract are features that are highly correlated with the flag in the dataset.

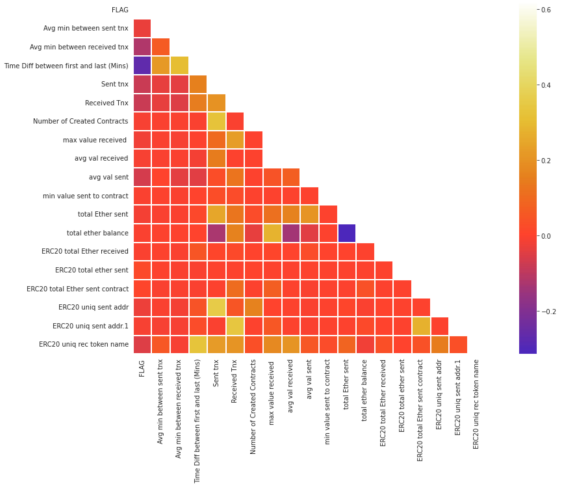

Figure 1: Correlation Matrix

4.3 Analytical Scoring

We added an analytical scoring model with the goal of enhancing the scoring of the other models by integrating a direct scoring approaches based on a mix of financial metrics and technical approaches. We used transaction analysis, NFT tracking, web of implied trust, KYC, and account metrics (Age, Balance, etc. . .) along with the possibility of other scoring and identity metrics along with user input(Reports). Metrics like NFT tracking and the web of implied trust aim to account for the anonymity in the DeFi by adjusting a wallets accountability based on the accountability of trading partners and an NFTs previous owners. These methods are adjusted on the fly and are able to customize the approach and address smaller issues.

4.4 Graph-based anomaly detection

In addition to the analytical and ML-based scores, we have augmented the Ethereum address scoring with graph-based anomaly detection approach. We use several graph-based outlier detection models to predict the probability of a node having anomalous transaction patterns. We assume that fraudulent network addresses would have a higher probability of being an outlier. However, this score should be used with caution as outlier probability is not directly related to risk of an address being

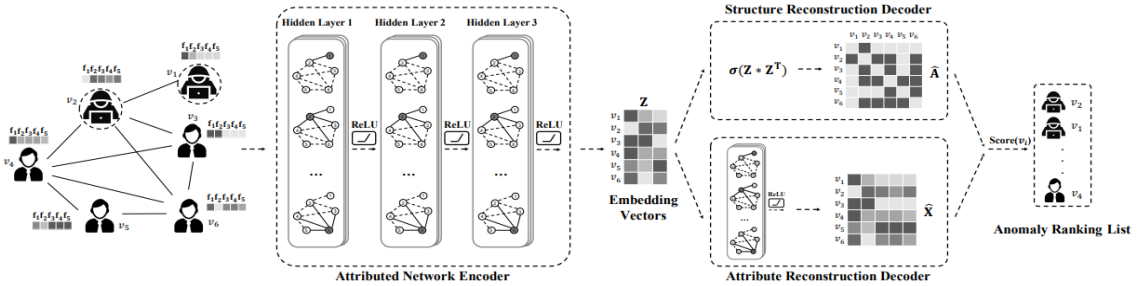

Figure 2: The overall architecture of the DOMINANT anomaly detection model [9]

fraudulent. Graph-based methods have been used for both fraud detection [10, 11, 12] and anomaly detection [9, 13, 14] in transaction networks. An example of a graph neural network architecture for anomaly detection is provided in Figure (Figure 2).

We have modeled the Ethereum blockchain network as a directed weighted multigraph, where nodes correspond to addresses, and edges represent the transaction values and timestamps. Due to computational resource constraints, for the purposes of this project, we sampled a small subgraph (|V|=100) composed of nodes randomly sampled from the Ethereum Fraud Detection Dataset [7] together with all their immediate neighbors. The blockchain timestamps were converted to Unix epochs. (Figure 3)

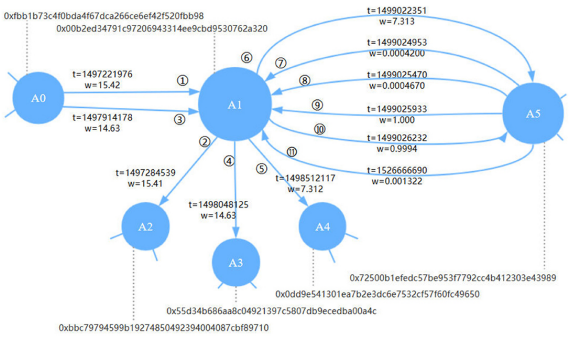

Figure 3: Ethereum network graph [15]

We used the PyGOD [16, 17] graph outlier detection library to train prediction models. Eleven outlier detection methods were trained on the sample subgraph: AdONE [18], ANOMALOUS [19], CoLA [20], CONAD [21], DOMINANT [9], DONE [18], GAAN [22], GCNAE[23], MLPAE[23], Radar[24], SCAN [25].

5 Evaluation

Based on the approaches outlined above, we study results by combining all three approaches. We built a web-app which, when given an address as input, displays the results from all three approaches individually and we also combine all three approaches to provide a score. This score is an overview and combination of scores from all three approaches, weighted. Currently, we have not conducted research on the most effective ways to weigh the different approaches, but it is clear from our research that the ML based approach would be weighed the most.[2]

5.1 ML models

We used 80:20 train test split on our large labeled ethereum datasets to evaluate the ML models, and we ran cross validation with k fold testing (k=10) to prevent overfitting. We ran Logistic Regression, Random Forest, XGB, and K-Nearest Neighbors classification models. The table shows a brief performance summary of ML models we trained. We measured Precision, Recall, and Accuracy metrics for all models. As can be seen in the table, the XGB classifier achieved the best results across all metrics.

We provide an average performance summary in (Table 1).

Table 1: ML Models

	Precision [Pos]	Recall [Pos]	Accuracy
Logistic Regression	0.68	0.88	0.89
Random Forest	0.93	0.95	0.97
XGB	0.94	0.96	0.98
K-Nearest Neighbors	0.86	0.95	0.96

[2] https://www.youtube.com/watch?v=zkLltxEhoxI

The Ethereum blockchain network is growing rapidly both in number of addresses and transactions, and changing its structural properties as new contracts are introduced. The supervised fraud detection datasets used in our work were created several years ago and do not capture the current state of the network. A practical implementation would require continous monitoring of the network for fraudulent addresses using a variety of sources and updating the training datasets for ML models.

5.2 Graph-based anomaly detection

For this project, we have implemented 11 node-level outlier detection algorithms over a subgraph of the Ethereum blockchain network using the PyGOD library. We have used a randomly sampled subset of nodes from the Ethereum Fraud Detection Dataset for our experiments. Most of the graph-based model implementations in the PyGOD library have a $\mathcal{O}(|V|^2)$ memory complexity, thus making training on the full dataset very expensive computationally. Most of the real-world blockchain networks have a very large number of nodes and edges, where a straightforward implementation is not feasible.

However, a more scalable solution, which is beyond the scope of this project, could be implemented directly using the Pytorch Geometric library which supports mini-batch sampling methods. Additionally, careful pruning of the network by removing non-fraudulent nodes with very high in- and/or out-degrees could bring down the memory and time complexity.

The implementation can be also extended for edge-level (transaction) and graph-level classification.

5.3 Analytical Scoring

We evaluated the analytical scoring by backtesting, for each component we backtest it on the fraud database to see how well each score lines up with the ground truth.

- NFT Tracking: Wallets who own NFTs that were previously owned by low accountability wallets are more likely to low accountability.
- Web of Implied Trust: Wallets that trade frequently with and/or send high amounts to low accountability individuals are more likely to low accountability.
- KYC: KYC highly correlated to higher accountability
- Account Metrics: Older and more active accounts with higher balances are more accountable.
- Other (Untested) Metrics: We are experimenting with fraud reporting and integrating other metrics (different Identity metrics and debt analysis).

Conclusions

Accountability in DeFi is a huge issue with no accepted standard solution. We proposed, designed, tested, and built a multi-modal solution based on traditional financial analysis metrics and cutting-edge tech. We made sure to target the key issues unique to DeFi like anonymity and lack of standard identity protocols. If there were more time to pursue this project, we would look into adding different metrics, tuning the system, and exploring the integration of new technologies like privacy respecting KYC.

References

[1] Bryan Ford. Identity and personhood in digital democracy: Evaluating inclusion, equality, security, and privacy in pseudonym parties and other proofs of personhood. https://arxiv.org/pdf/2011.02412.pdf.

[2] Stifter N. Kost'al K. Saglam C. Sabadello M. Fdhila, W. Methods for decentralized identities: Evaluation and insights. bpm, 2021. https://eprint.iacr.org/2021/1087.pdf.

[3] D. Maram et al. Candid: Can-do decentralized identity with legacy compatibility, sybil-resistance, and accountability. *IEEE Symposium on Security and Privacy (SP)*, pages 1348–1366, 2021.

[4] Web of trust. https://en.wikipedia.org/wiki/Web_of_trust.

[5] How to invest in defi using the 5 key performance indicators (kpi). https://learn.bybit.com/defi/investing-in-defi-using-key-performance-indicators/.

[6] Scoring protocol risk. https://github.com/ConsenSys/defi-score.

[7] Ethereum fraud detection dataset. https://www.kaggle.com/datasets/vagifa/ethereum-frauddetection-dataset.

[8] Ethereum fraud dataset. https://www.kaggle.com/datasets/gescobero/ethereum-fraud-dataset.

[9] Kaize Ding, Jundong Li, Rohit Bhanushali, and Huan Liu. Deep anomaly detection on attributed networks. In *SDM*, 2019.

[10] Hiroki Kanezashi, Toyotaro Suzumura, Xin Liu, and Takahiro Hirofuchi. Ethereum fraud detection with heterogeneous graph neural networks. *ArXiv*, abs/2203.12363, 2022.

[11] Jiajing Wu, Qi Yuan, Dan yan Lin, Wei You, Weili Chen, Chuan Chen, and Zibin Zheng. Who are the phishers? phishing scam detection on ethereum via network embedding. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 52:1156–1166, 2019.

[12] Liang Chen, Jiaying Peng, Yang Liu, Jintang Li, Fenfang Xie, and Zibin Zheng. Phishing scams detection in ethereum transaction network. *ACM Transactions on Internet Technology (TOIT)*, 21:1 – 16, 2020.

[13] Xiaoxiao Ma, Jia Wu, Shan Xue, Jian Yang, Quan Z. Sheng, and Hui Xiong. A comprehensive survey on graph anomaly detection with deep learning. *ArXiv*, abs/2106.07178, 2021.

[14] Kaize Ding, Jundong Li, and Huan Liu. Interactive anomaly detection on attributed networks. *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining*, 2019.

[15] Jiajing Wu, Dan yan Lin, Zibin Zheng, and Qi Yuan. T-edge: Temporal weighted multidigraph embedding for ethereum transaction network analysis. In *Frontiers of Physics*, 2019.

[16] Kay Liu, Yingtong Dou, Yue Zhao, Xueying Ding, Xiyang Hu, Ruitong Zhang, Kaize Ding, Canyu Chen, Hao Peng, Kai Shu, George H. Chen, Zhihao Jia, and Philip S. Yu. Pygod: A python library for graph outlier detection. *arXiv preprint arXiv:2204.12095*, 2022.

[17] Kay Liu, Yingtong Dou, Yue Zhao, Xueying Ding, Xiyang Hu, Ruitong Zhang, Kaize Ding, Canyu Chen, Hao Peng, Kai Shu, Lichao Sun, Jundong Li, George H. Chen, Zhihao Jia, and Philip S. Yu. Bond: Benchmarking unsupervised outlier node detection on static attributed graphs. *arXiv preprint arXiv:2206.10071*, 2022.

[18] Sambaran Bandyopadhyay, N. Lokesh, Saley Vishal Vivek, and M. Narasimha Murty. Outlier resistant unsupervised deep architectures for attributed network embedding. *Proceedings of the 13th International Conference on Web Search and Data Mining*, 2020.

[19] Zhen Peng, Minnan Luo, Jundong Li, Huan Liu, and Qinghua Zheng. Anomalous: A joint modeling approach for anomaly detection on attributed networks. In *International Joint Conference on Artificial Intelligence*, 2018.

[20] Yixin Liu, Zhao Li, Shirui Pan, Chen Gong, Chuan Zhou, and George Karypis. Anomaly detection on attributed networks via contrastive self-supervised learning. *IEEE Transactions on Neural Networks and Learning Systems*, 33:2378–2392, 2021.

[21] Zhiming Xu, Xiao Huang, Yue Zhao, Yushun Dong, and Jundong Li. Contrastive attributed network anomaly detection with data augmentation. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, 2022.

[22] Zhenxing Chen, Bo Liu, Meiqing Wang, Peng Dai, Jun Lv, and Liefeng Bo. Generative adversarial attributed network anomaly detection. *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, 2020.

[23] Xu Yuan, Na Zhou, Shuo Yu, Huafei Huang, Zhikui Chen, and Feng Xia. Higher-order structure based anomaly detection on attributed networks. *2021 IEEE International Conference on Big Data (Big Data)*, pages 2691–2700, 2021.

[24] Jundong Li, Harsh Dani, Xia Hu, and Huan Liu. Radar: Residual analysis for anomaly detection in attributed networks. In *International Joint Conference on Artificial Intelligence*, 2017.

[25] Xiaowei Xu, Nurcan Yuruk, Zhidan Feng, and Thomas A. J. Schweiger. Scan: a structural clustering algorithm for networks. In *Knowledge Discovery and Data Mining*, 2007.